Subject: LVDS tsmc process. simulation

File: O:\projects\Lvds\Albert Chang\tsmc_ans.txt 2000/2/11, 02:50:43PM

Subject: [Fwd: Questions about 0.35um Logic SPICE model again]
Date: Fri, 11 Feb 2000 14:13:09 +0800
From: Chien-Jian Tseng <cj.tseng@trumpion.com.tw>
Organization: Trumpion Microelectronics Inc.
To: 鄭 建勳 <jensen.cheng@trumpion.com.tw>,ray07010@tomail.com.tw,
peter.hsu@trumpion.com.tw,blair.chen@trumpion.com.tw

Subject: Re: Questions about 0.35um Logic SPICE model again
Date: Fri, 11 Feb 2000 13:05:35 +0800
From: "CYTUNG" <cytung@tsmc.com.tw>
To: cj.tseng@trumpion.com.tw

Hi C.J.:
Sorry to answer your questions. I put my answers in your question list.

Sorry to bother you again. About how to use 0.35um Logic Silicide SPICE model correctly, I still have some questions as followed:

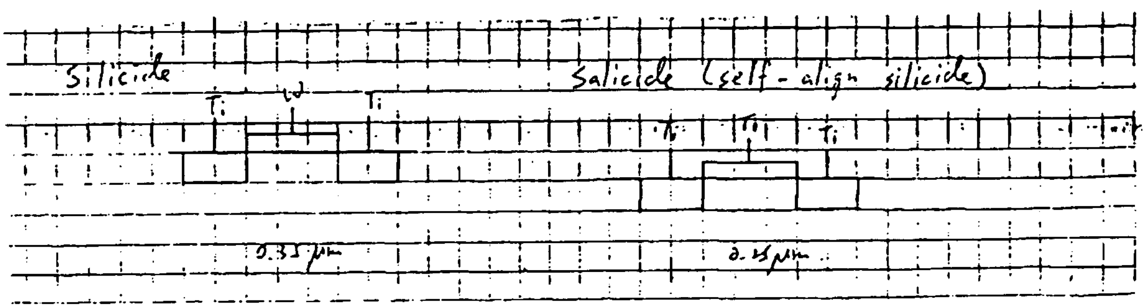
1. for poly & p+ diff. resistor value, the variance is within 6 sigma.
Is it possible to control the variance within 3 sigma?
Ans: We can't use 3 sigma to our SPEC, it will increase our wafer scrap rate. But basically, our control can locate within 3 sigma. The variance locates out 3 sigma, the chance is very low.
2. What's the current density for poly resistor? I mean what's the max. current that a 1um wide poly can afford? and for NW resistor? for p+diff. resistor?
Ans: We don't have this kind of data.
3. for p+NW diode model (PDIO). When I do the diode layout, are there any specific design rules related to these p+ diff. and NW?
Ans: There is no special rule on the Dio design except the process design rule. However, we recommend the diode layout should be in cubic style since that will get more accurate simulation result.
4. I notice that for 0.25um process, that's a "salicide" process instead of "silicide" process. I am not quite familiar with semiconductor process. Could you briefly explain what's the different between "salicide" and "silicide" process?
Ans: silicide means source/drain area used Ti silicide, but gate area used W silicide. We used two different steps to get them. -->0.35um process
Salicidie mean self-align silicide. It means souce/drain/gate area use Ti silicide. We used one steps to get them. -->0.25um process
I appreciate your kindly help on those issues, and happy lunar new year.
--

曾千健
IC Design Engineer

創品電子
http://www.trumpion.com

12F-5, No. 17, Sec. 1, Cheng-Teh Rd.
Taipei, Taiwan, R.O.C.
Tel:886-2-25587855 ext. 316
Fax:886-2-25587850
Net:cj.tseng@trumpion.com.tw

EXHT. # 12
PLTF/DEF
WIT. ALBERT CHANG
DATE 2/25/03
SHERRY SHERRY, CSR 6819



Disclosed To And Understood By Me

Name Date

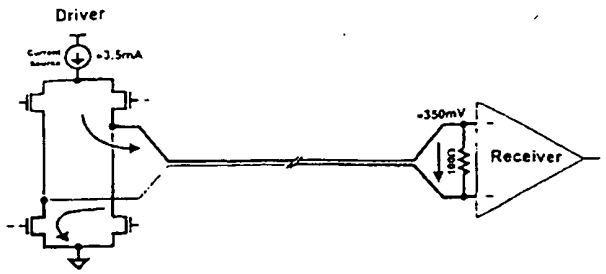
Name Date

Signature: blair

Date: 2000/2/11

17002

Subject: LVDS concept



Simplified diagram of LVDS driver and receiver connected via 100Ω controlled differential impedance media.

receiver ~ 350mV peak to peak

line 1 → ← line 2

sclosed To And Understood By Me

Name Date

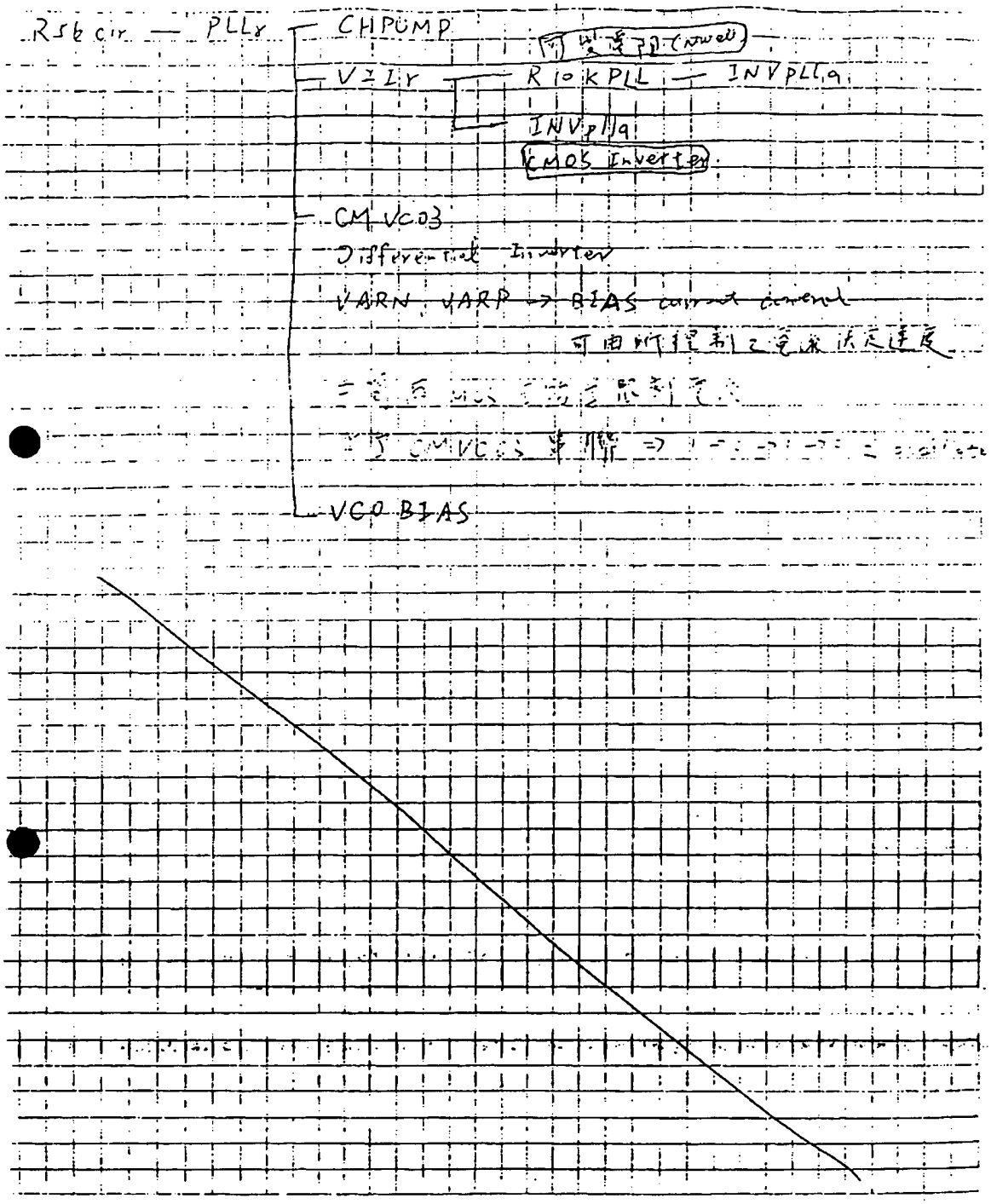
Name Date

Signature: blair

Date: 10/3/12

Subject: LVDS final (0307)

17003



Disclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 2003/3/13

17004

Subject: PLL study

Analog PLL 는 $f_o - f_i \Rightarrow$ voltage $\rightarrow$ control VCO

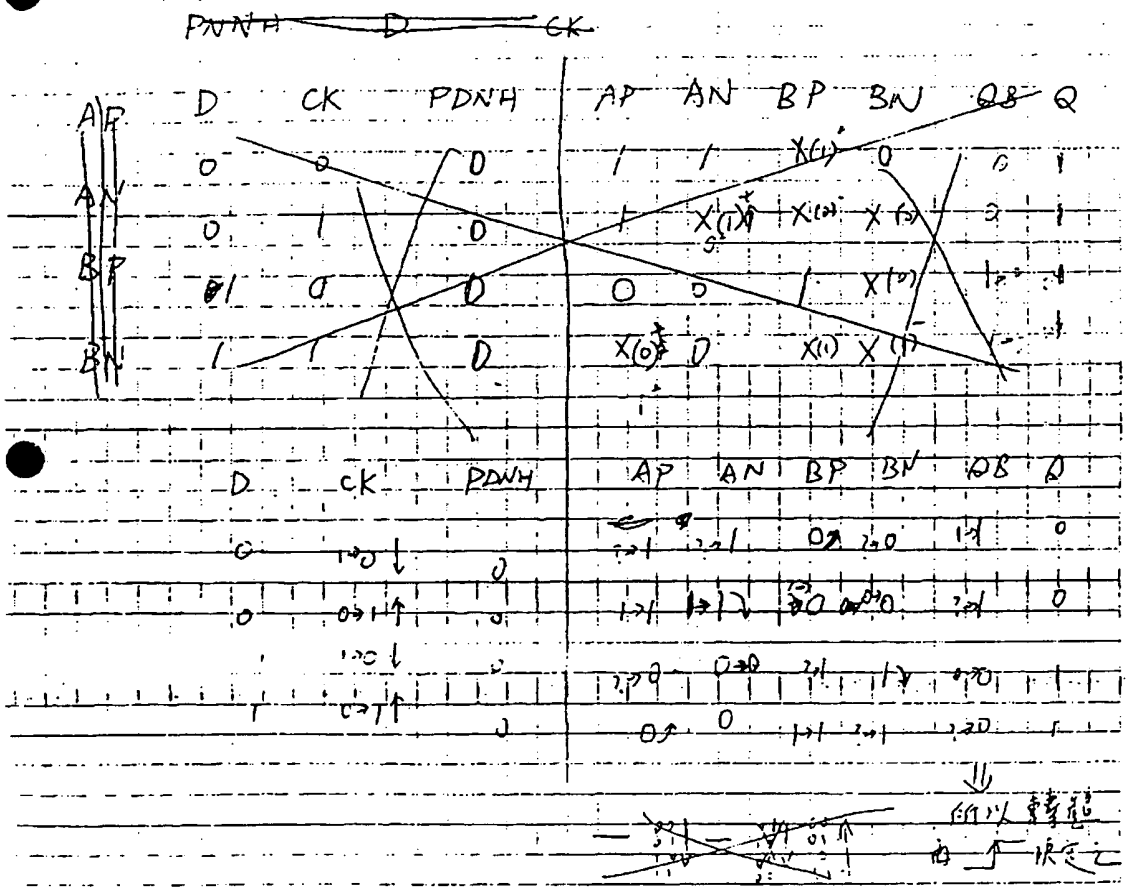
necessary control voltage $\rightarrow V_c$

PD output $\Rightarrow V_d$

$(\theta_i - \theta_o) = \theta_d$

no input $\Rightarrow \theta_{d_0} \Rightarrow V_{d_0}$

1 SPDFFd

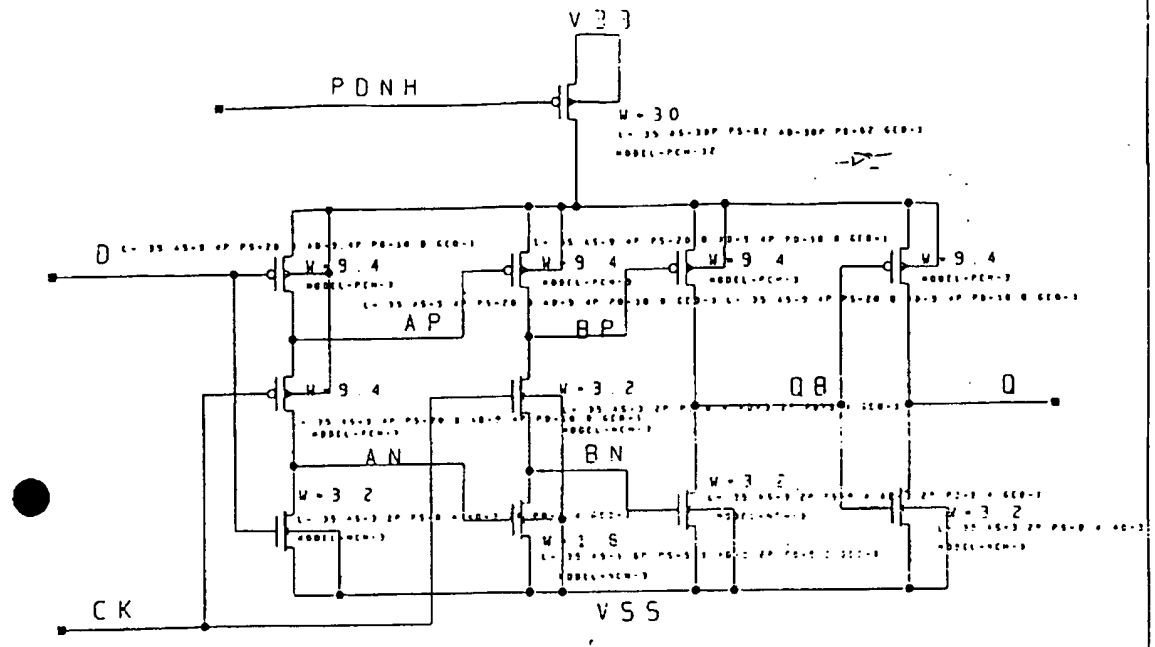


Disclosed To And Understood By Me

Name Date Signature: blair

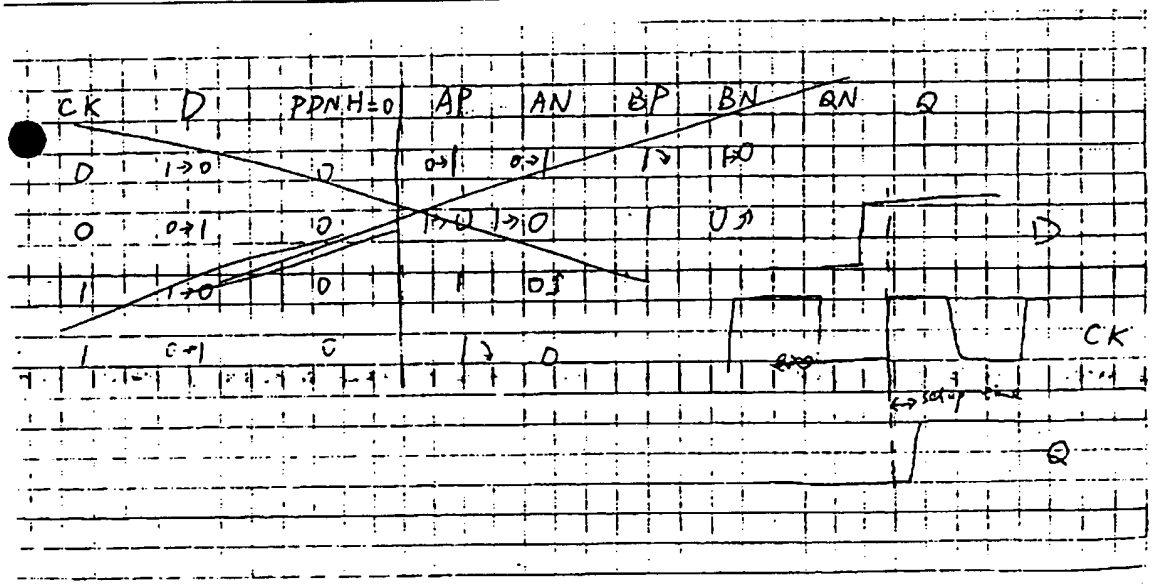
Name Date Date: 2022/3/16

Subject: _____



TSPDFFd

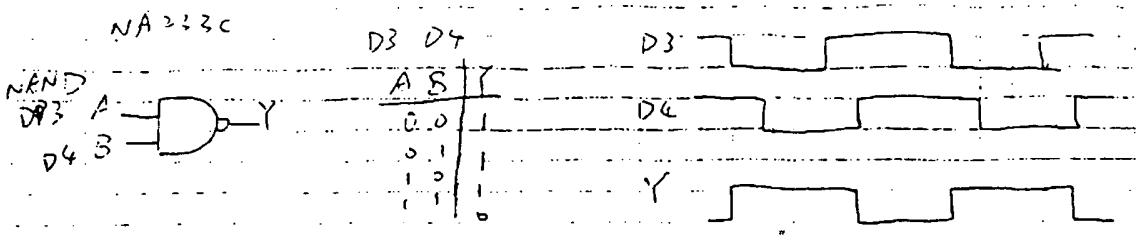
SQ by Albert.Yi long Chang
Date: Aug.08.1999

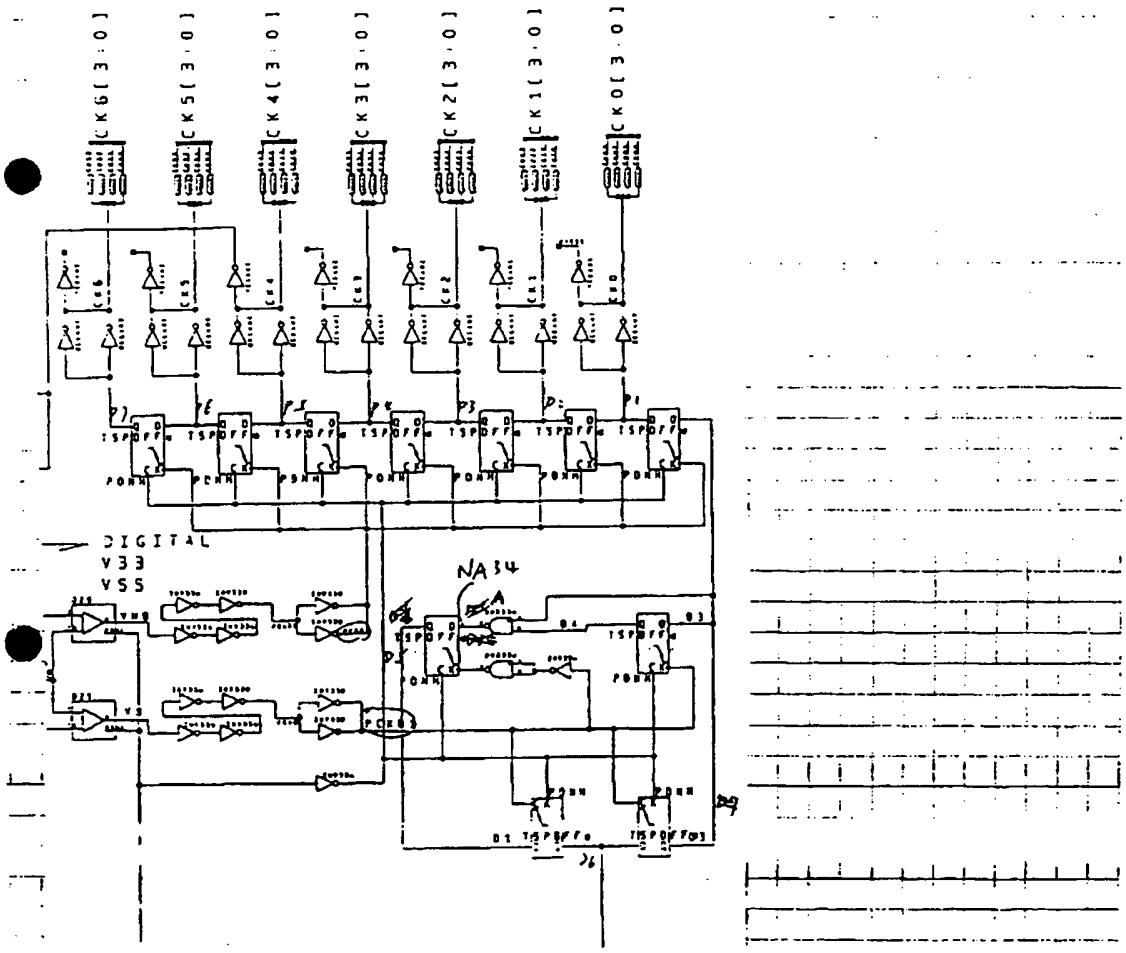


Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/3/16

Subject: 17006 PLL study - 7 phase loop





sclosed To And Understood By Me

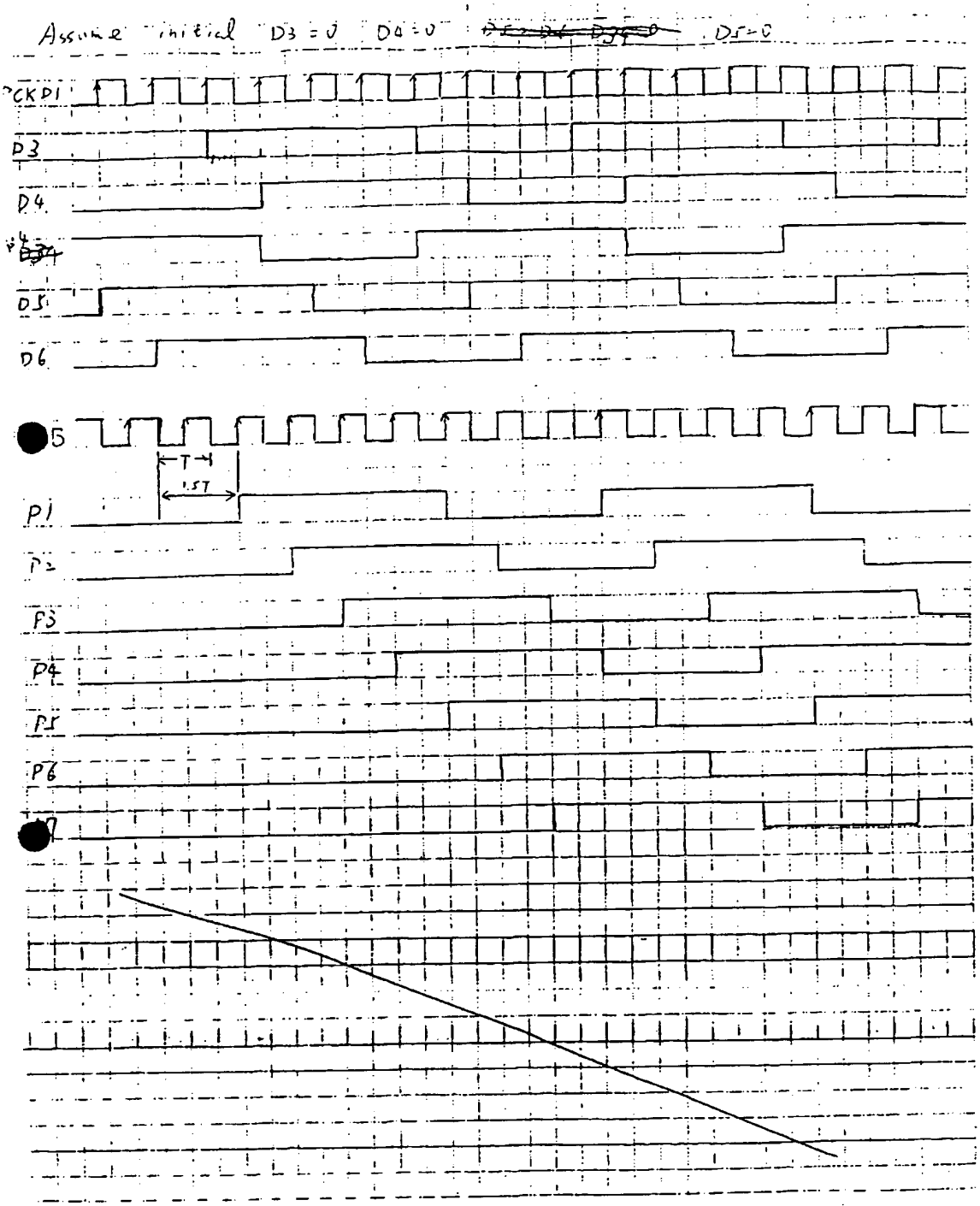
Name Date

Name Date

Signature: blair

Date: 2003/5/2

Subject: _____



Disclosed To And Understood By Me

_____ Name _____ Date

_____ Name _____ Date

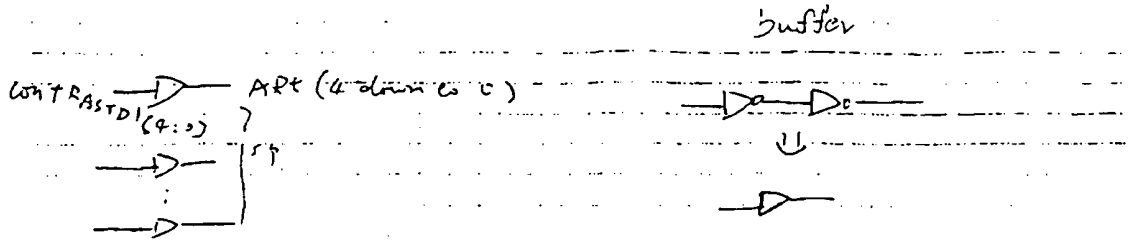
Signature: _____

Date: 22-7/3/20

1/0000

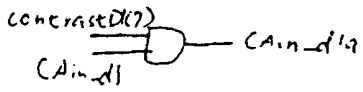
Subject: 0s/0 VHDL code study

ARt ⇐ CONTRASTD1 (4 downto 0);



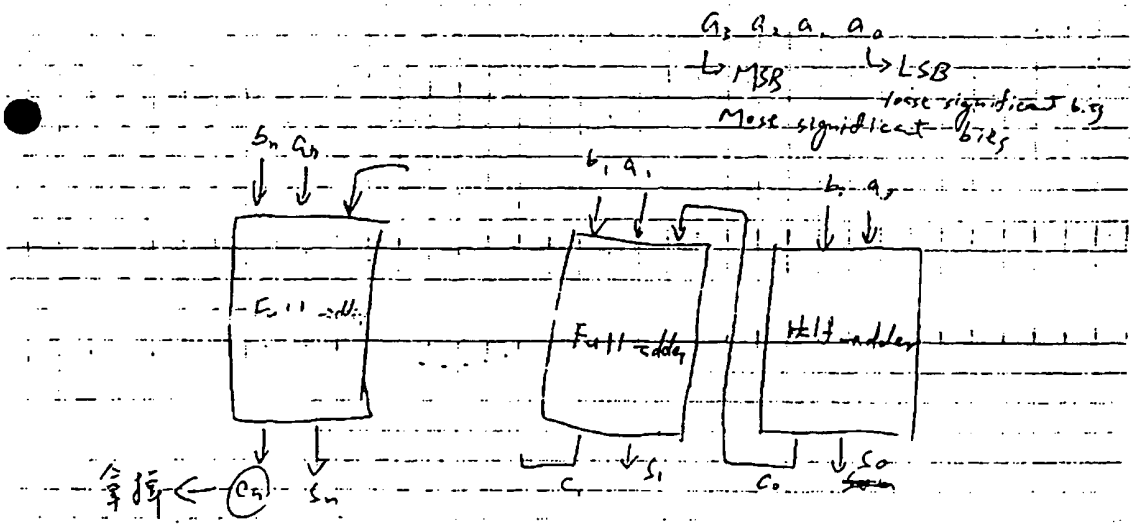
CAin_d1a ⇐ (others ⇒ '0') when CONTRASTD1(7)='0'

else CAin_d1;



CAin_d2a ⇐ CAin_d2 + Ain_d2

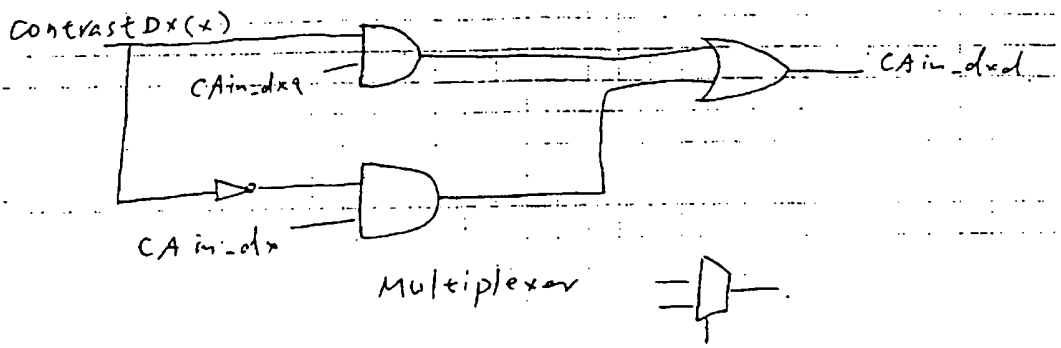
CAin_d3a ⇐ CAin_d3 + Ain_d3



isclosed To And Understood By Me

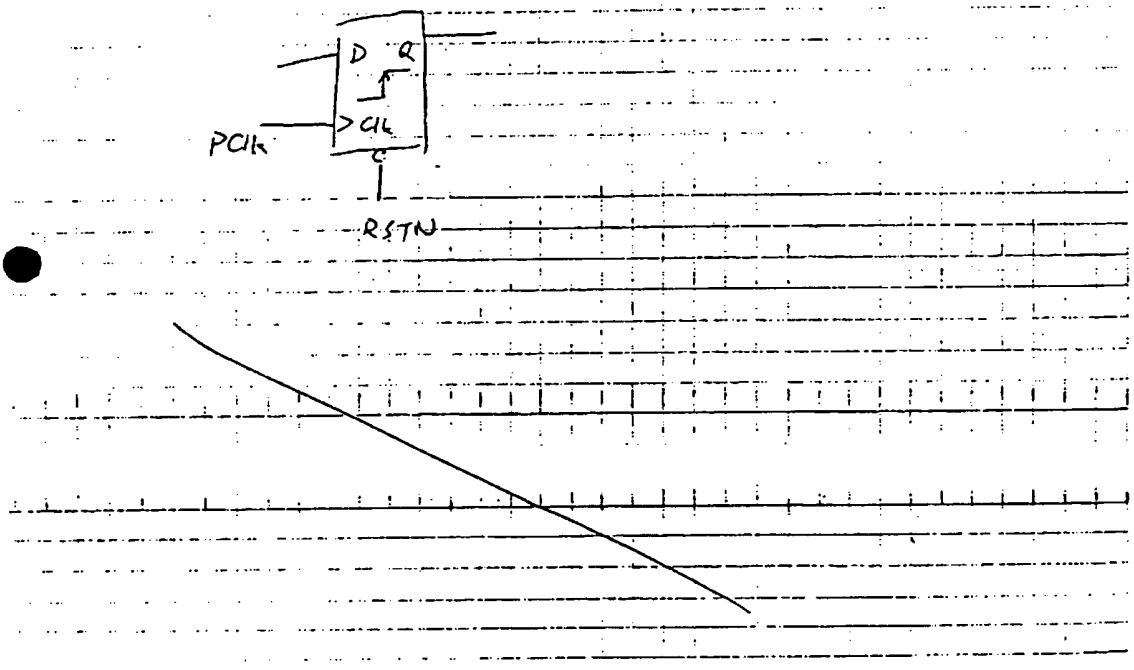
Name	Date	Signature: blair
Name	Date	Date: 2003/3/m

CAin_dxd ⇐ CAin_dxq when contrastDx(x)='1' else CAin_dx



IF (PClk'event AND PClk='1') THEN

IF (RSTN='0' or RSTN='L') THEN ELSE



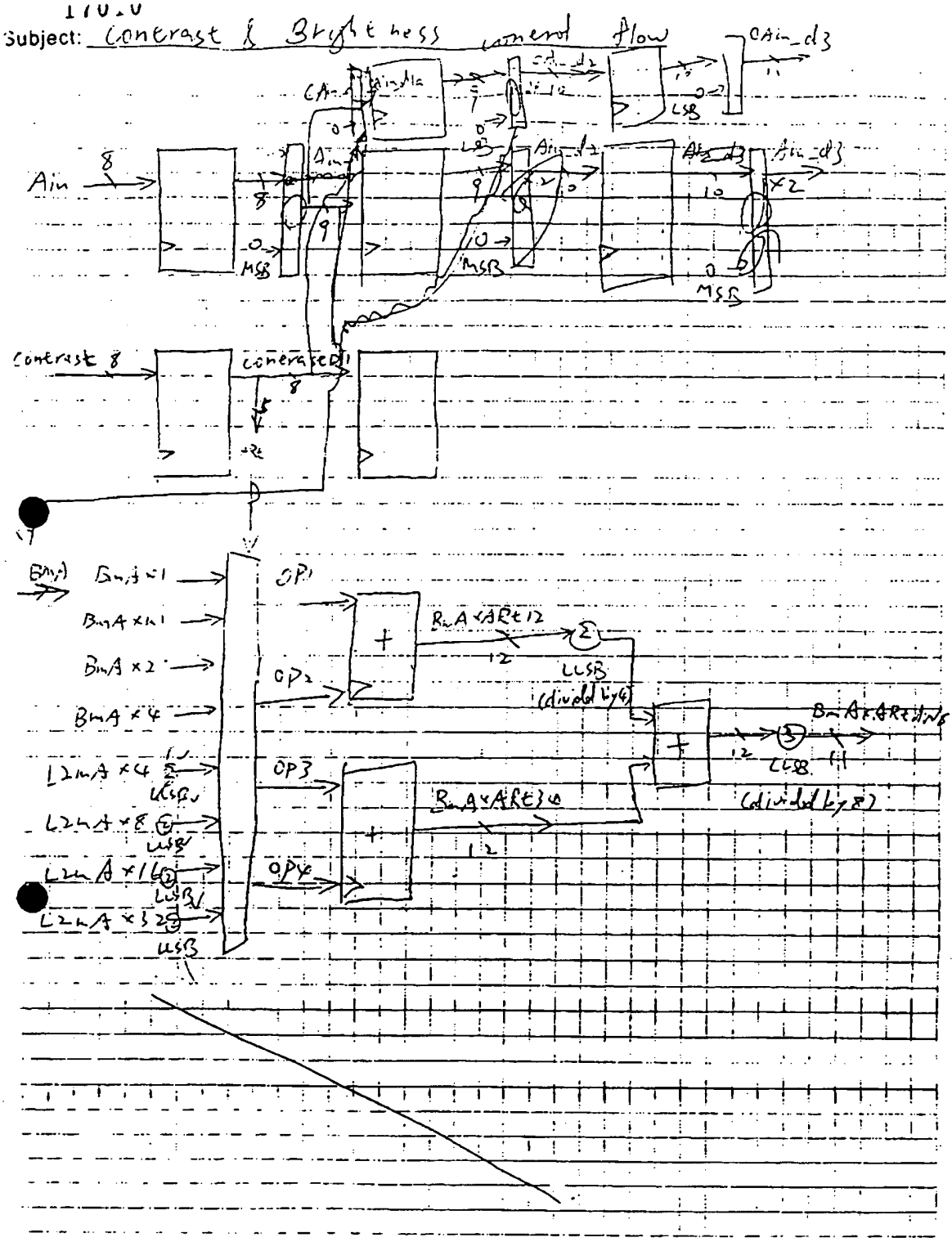
Disclosed To And Understood By Me

Name Date Signature:

Name Date Date:

110.0

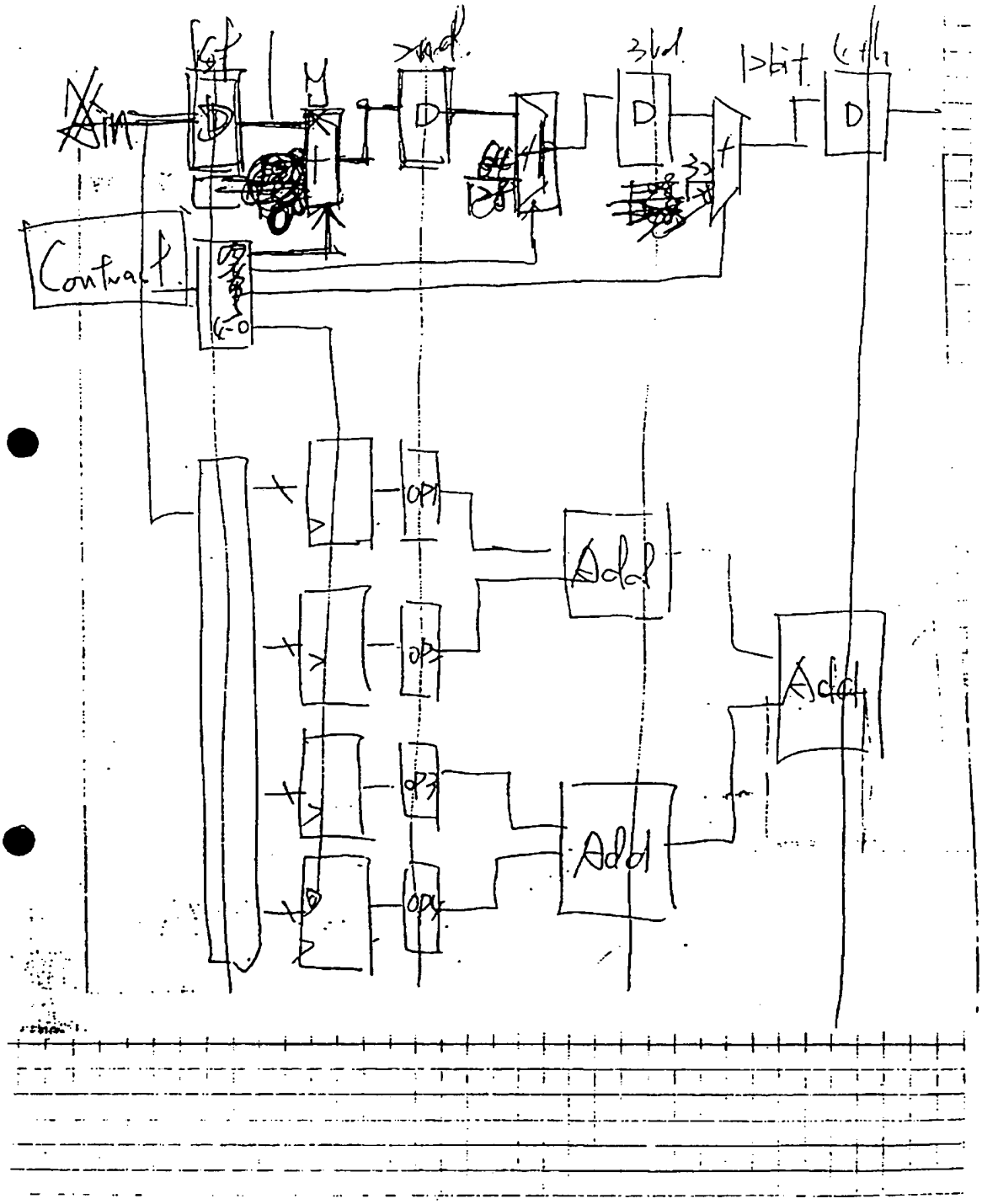
Subject: Contrast & Brightness control flow



sclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2020/3/

Subject: _____ 17011

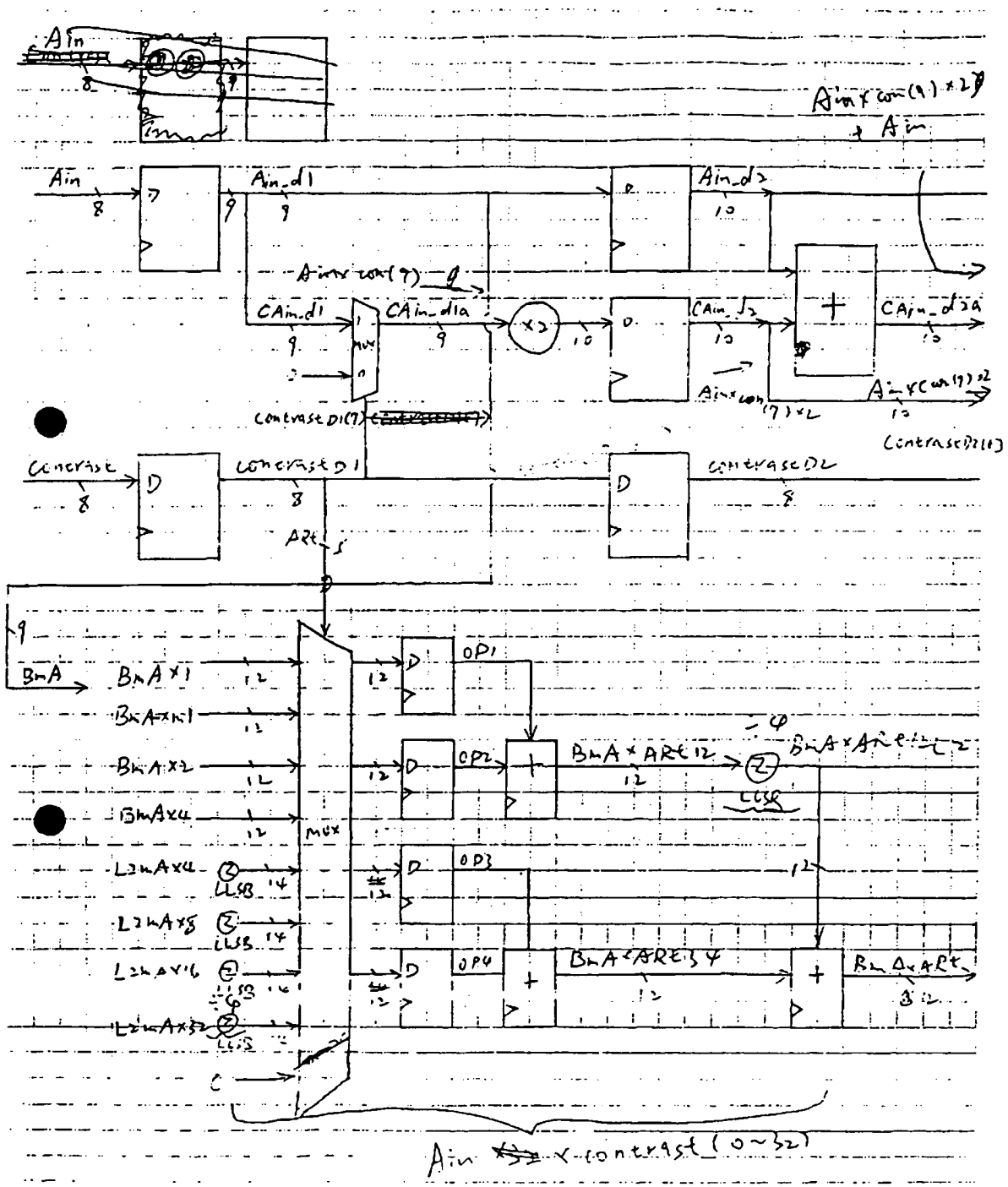


Disclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 2005/3/12

17012

Subject: _____



Disclosed To And Understood By Me

_____ _____

Name Date

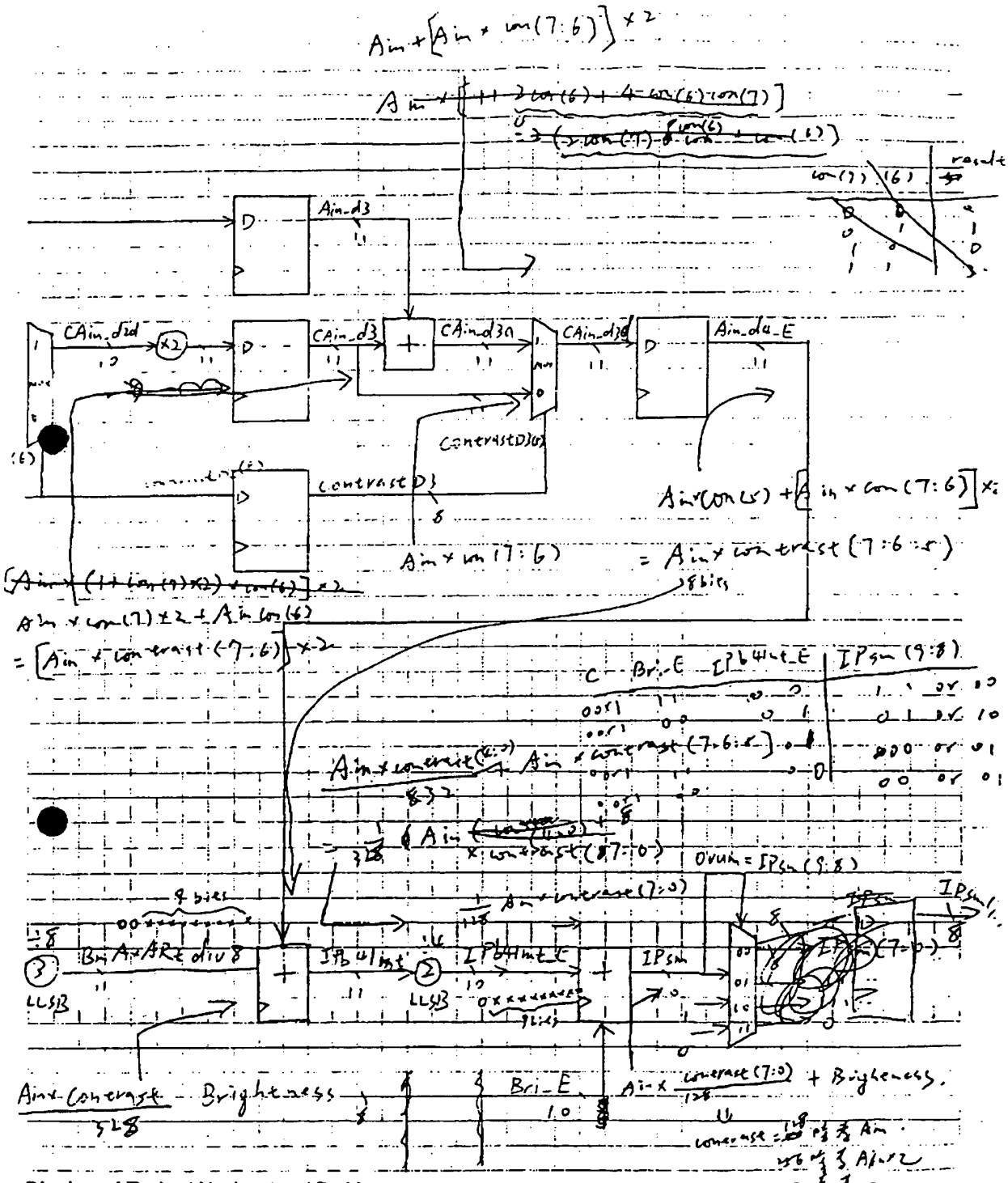
Signature: blair

_____ _____

Name Date

Date: 2003/3/12

Subject: _____



Disclosed To And Understood By Me

Name	Date	Signature:	
Name	Date	Date:	2000/3/22

17014

Subject: Meaning of BmA×m1

BmA×m1 = (NOT (BmA×1)) + 1

BmA×1 = BmA(7) & BmA(7) & BmA(7) & BmA;

Assume BmA = 0000 0001

=> BmA×1 = 000000000001

=> BmA×m1 = 111111111110 + 1

= 111111111111 (complement of BmA×1)

Assume BmA = 00100000

=> BmA×1 = 000000100000

=> BmA×m1 = 111111011111 + 1

= 111111100000 (complement of BmA×1)

Assume BmA = 11000010

=> BmA×1 = 111111000010

=> BmA×m1 = 000000111101 + 1

= 000000111110 (complement of BmA×1)

=> BmA×m1 = complement of BmA×1

BmA×m1 的 MSB 之後 = complement of BmA

Disclosed To And Understood By Me

Name Date

Name Date

Signature: ~~2003/3/27~~ blair

Date: 2003/3/27

17015

Subject: Brightness & Contrast

Contrast 0 ~ 255 ; C → 每個 = Brightness

128 → Ain + Brightness

255 → ~2 × Ain + Brightness

$IPsm = Am \times \frac{contrast}{128} + Brightness$

Brightness 為 2's complement

Brightness = 0 x x x x x x x ⇒ 正值

1 x x x x x x x ⇒ 負值

-128 ~ 127

Brightness < -128 (or -127) ⇒ IPsm 每一個 clk 出現 Brightness 值

$Ain \times \frac{contrast}{128} + Brightness < 0$

⇒ $Brightness < -Am \times \frac{contrast}{128}$ 時 IPsm = 0

⇒ $-Brightness > Ain \times contrast/128$

⇒ $|-128 \times Brightness| > Ain \times contrast$ 時 IPsm = 0

即 Brightness 為負數時

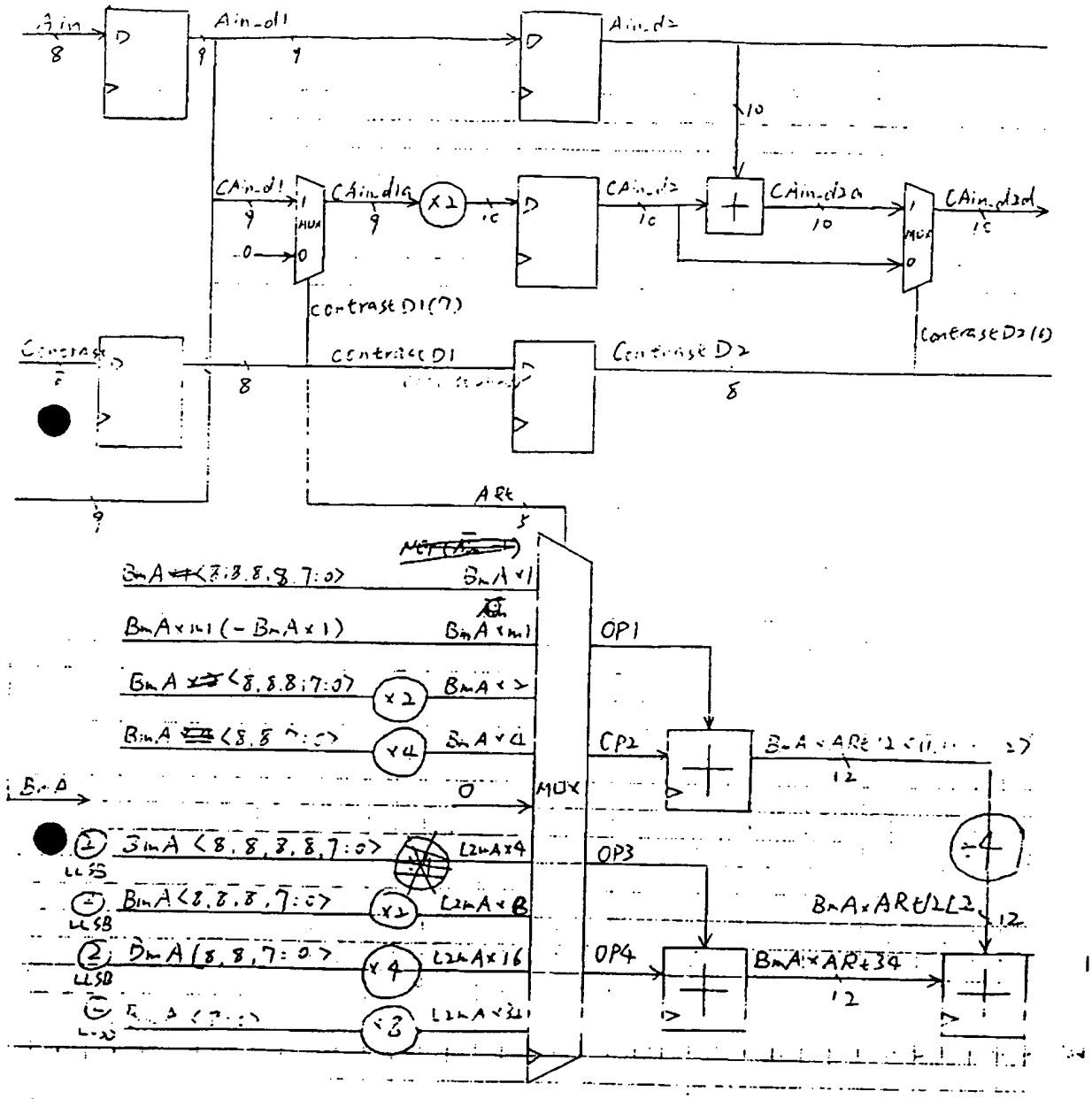
Disclosed To And Understood By Me

Name Date Signature: blin

Name Date Date: 2001/3/15

1/0/0

Subject: Contrast & Brightness Control Flow

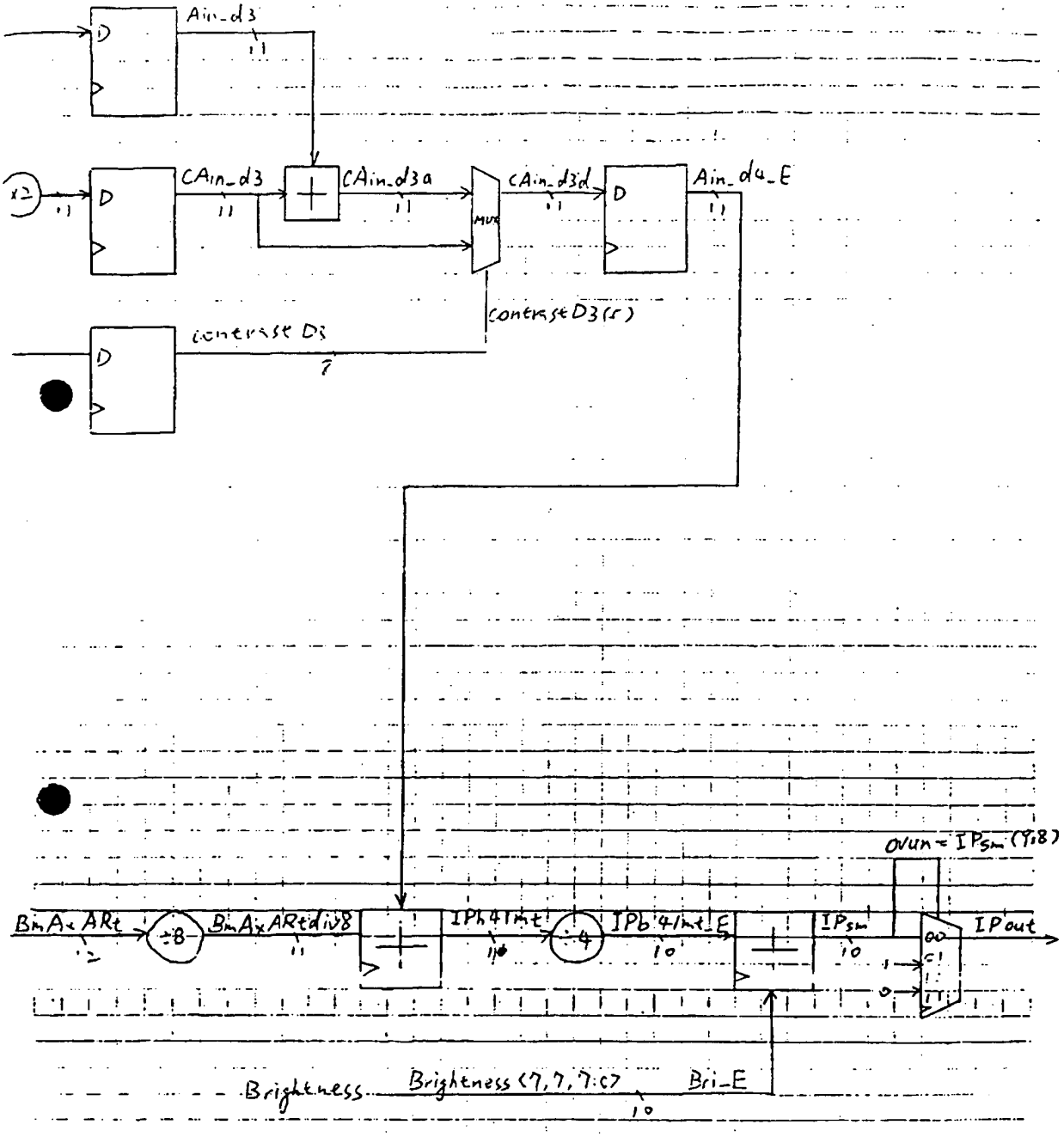


Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2/3/04

Subject: _____ 17017



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2003/3/24

17018

Subject: Contrast & Brightness YUV design

2.8 Color adjust, color space convert (matrix), and gamma

The OSLO chip performs color space conversion before color adjust and gamma after.

The YUV adjustments are

Yout = (Y-Yblack)*contrast + brightness

Uout = (U*cos(hue) + V*sin(hue))*saturation = U*sat_coshue + V*sat_sinhue

Vout = (V*cos(hue) - U*sin(hue))*saturation = V*sat_coshue - U*sat_sinhue

The RGB adjustments are

Rout = R*contrast + brightness

Gout = G*contrast + brightness

Bout = B*contrast + brightness

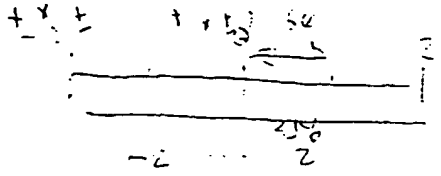


TABLE 3.

sat_sinhue correction for MSB=0						
sat_sinhue[7:0]	7f(hex)	7e	.....	02	01	00
multiply value	127/64	126/64	.....	2/64	1/64	0/64
sat_sinhue correction for MSB=1						
sat_sinhue[7:0]	ff(hex)	fe	.....	82	81	80
multiply value	-1/64	-2/64	.....	-126/64	-127/64	-128/64

sat_coshue correction for MSB=0						
sat_coshue[7:0]	7f(hex)	7e	.....	02	01	00
multiply value	127/64	126/64	.....	2/64	1/64	0/64
sat_coshue correction for MSB=1						
sat_coshue[7:0]	ff(hex)	fe	.....	82	81	80
multiply value	-1/64	-2/64	.....	-126/64	-127/64	-128/64

Yblack correction for MSB=0							
Yblack[7:0]	7f(hex)	7e	7d	.....	02	01	00
offset value	127	126	125	.....	2	1	0
Yblack correction for MSB=1							
Yblack[7:0]	ff(hex)	fe	fd	.....	82	81	80
offset value	-1	-2	-3	.....	-126	-127	-128

TABLE 4.

Contrast correction for MSB=0						
contrast[7:0]	7f(hex)	7e	.....	02	01	00
multiply value	255/128	254/128	.....	130/128	129/128	128/128
Contrast correction for MSB=1						

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 10/5/00

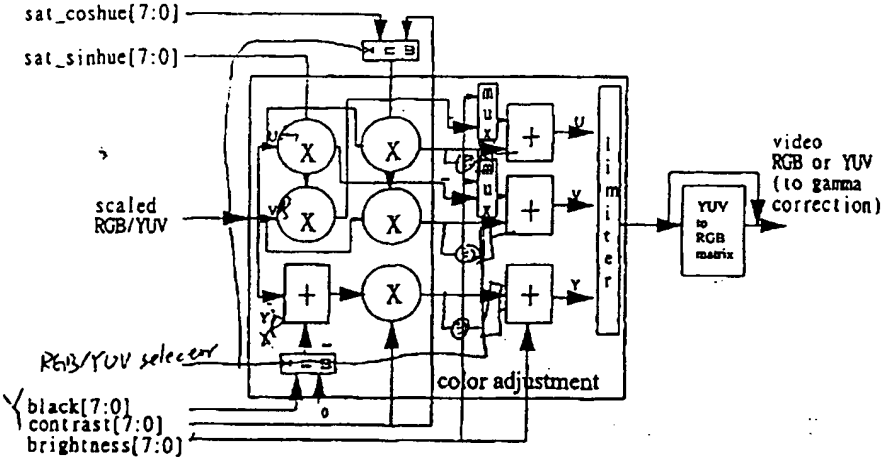
TABLE 4.

contrast[7:0]	ff(hex)	fe	.....	82	81	80
multiply value	127/128	126/128	.....	2/128	1/128	0/128

Brightness correction for MSB=0							
brightness[7:0]	7f(hex)	7e	7d	.....	02	01	00
offset value	127	126	125	.....	2	1	0
Brightness correction for MSB=1							
brightness[7:0]	ff(hex)	fe	fd	.....	82	81	80
offset value	-1	-2	-3	.....	-126	-127	-128

The block diagram is depicted as follows:

FIGURE 11. Functions of matrix and color adjustments for YUV and RGB



Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2003/3/24

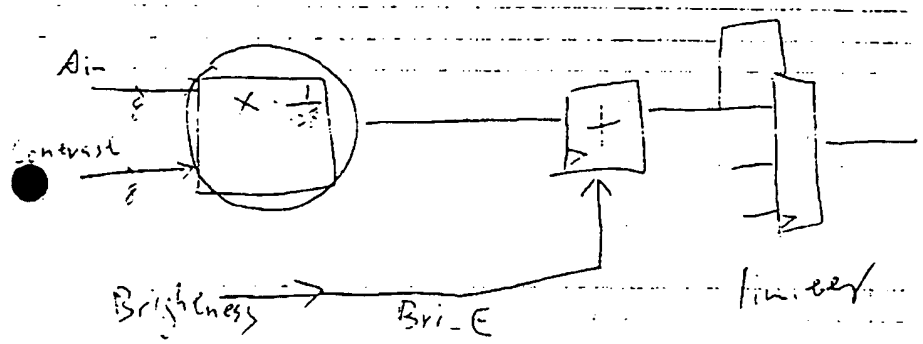
17020

Subject: _____

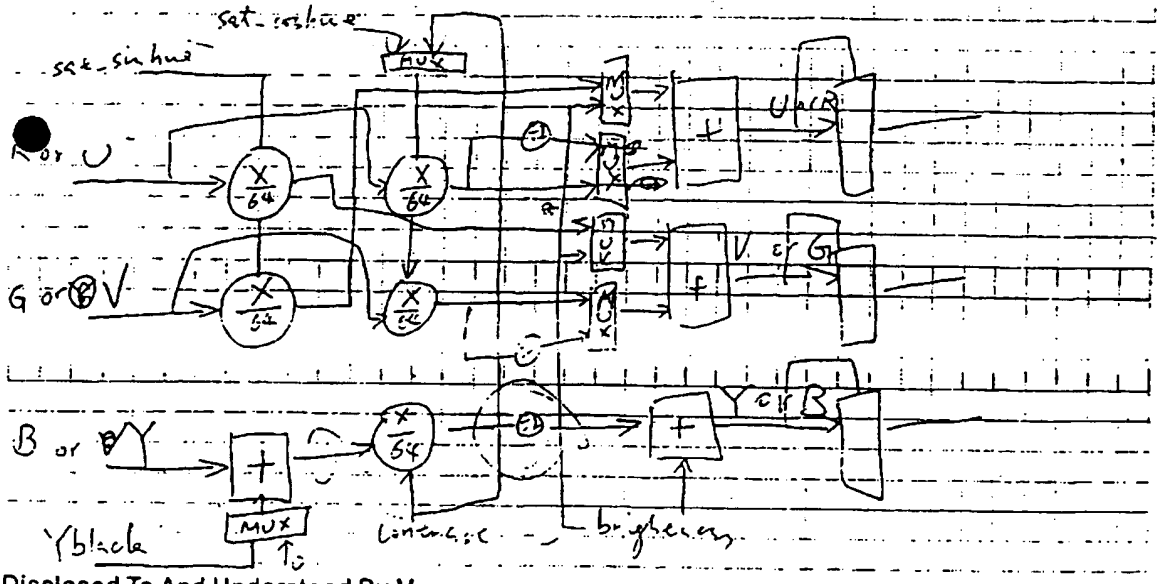
(X) 用(64) R.G.B 8 = 128 ... D = 2

Y.U.V

- 用 Zurac 之 Contrast & Brightness 可簡化多



In praceica . ÷2 式 x1 由 MUX 取代 .

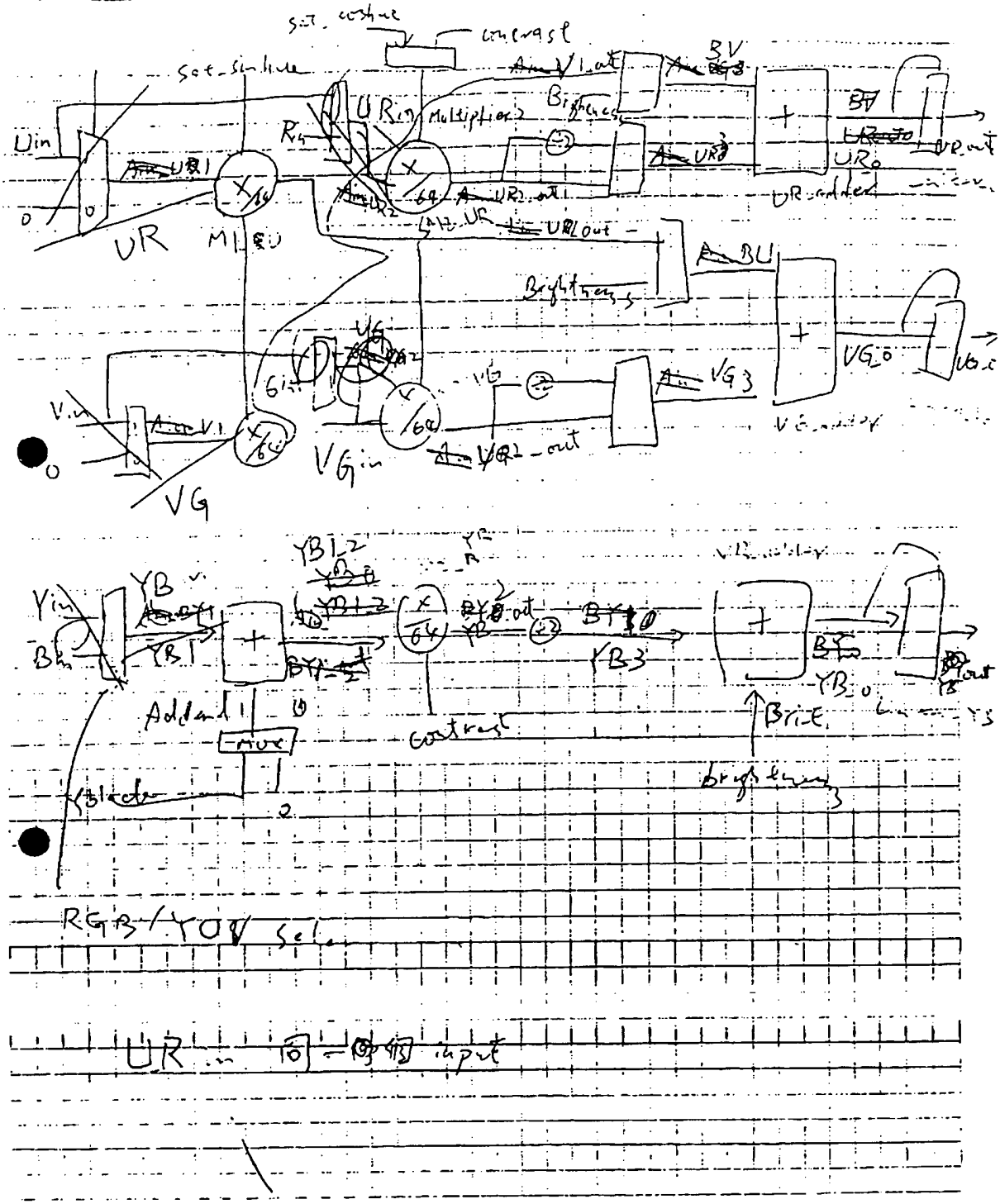


Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2005/4/26

Subject: _____

17021



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2002/3/27

17022

Subject: Concrase x Ain study

BnA ∈ Ain

~~BnAx1 = BnA<8.8.88.07 = NOT(Ain - 1)~~

BnAxn1 = B/NOT(BnA x1)) + 1 = $\overline{A_{in}}$

BnAx2 = BnA(8,8,8,0) & 0 = NOT($\overline{A_{in}}$ - 1) x2

BnAx4 = BnA(8,8:0) & 00 = NOT($\overline{A_{in}}$ - 1) x4

} OP1 + OP2

● B L2nAx4 = BnA(8.8.8.8.0) = NOT ($\overline{A_{in}}$ - 1)
= BnA x 1

L2nAx8 = BnA(8.8:8:0) & 0 = NOT($\overline{A_{in}}$ - 1) x2
= BnAx2

L2nAx16 = BnA(8,8:0) & 00 = NOT($\overline{A_{in}}$ - 1)x4
= BnAx4

} OP3 + OP4

● ~~L2nAx32 = BnA & 000 = BnA~~ Ain x 8

OP1 + OP2
= BnAxARE12L2 = BnAxARe(2(1+1+1+1-2)

= NOT(BnAXARE12 - 1)/4

sclosed To And Understood By Me

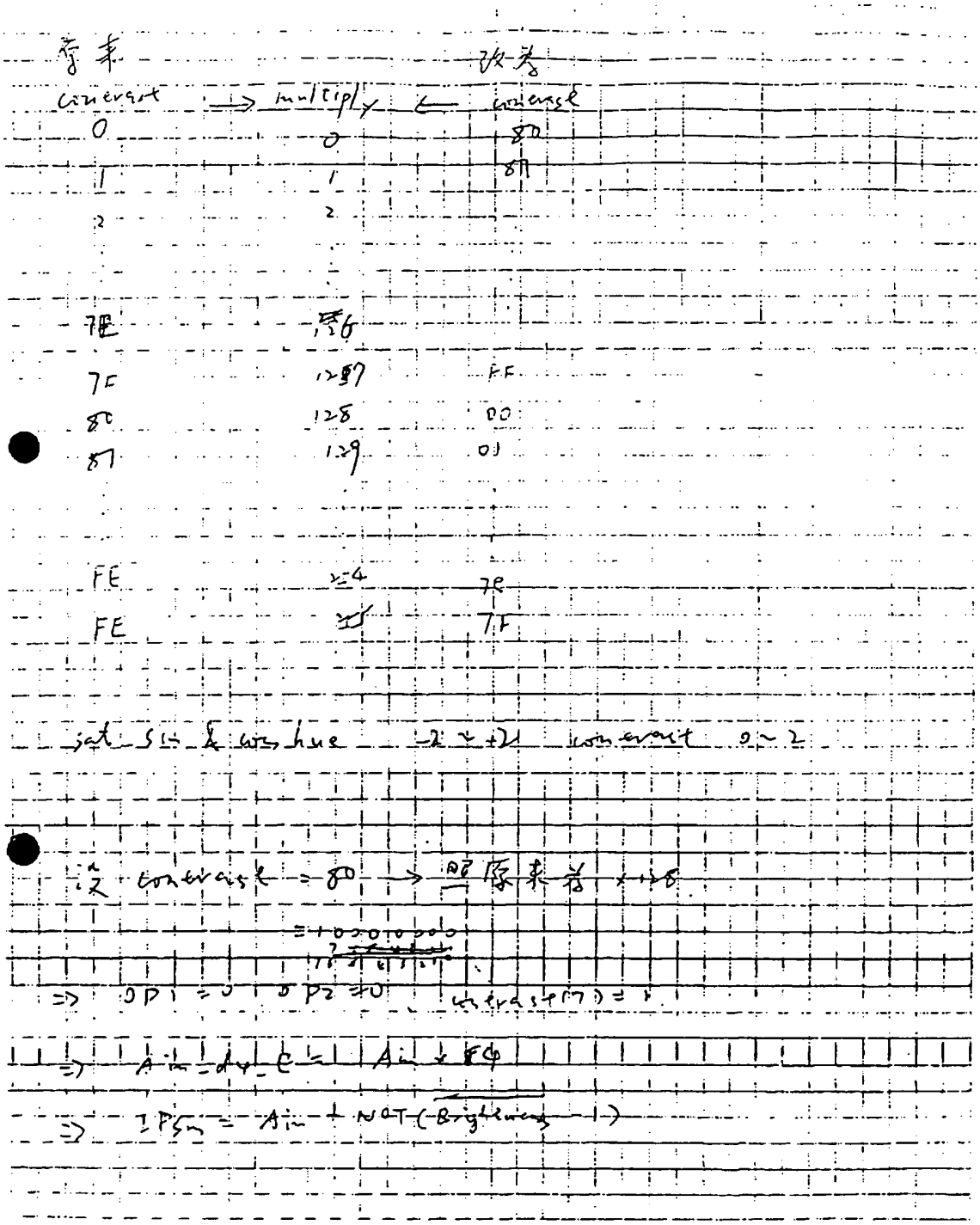
Name Date

Name Date

Signature: [signature]

Date: 2006/3/28

Subject: _____ 17023



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: [illegible]

Subject:

設 若 $AR_i = 11101 = 29$

$\Rightarrow OP1 = BnA \times 1 = NOT(\overline{A_{in}} - 1)$

$OP2 = BnA \times 4 = NOT(\overline{A_{in}} - 1) \times 4$

$OP3 = L2nA \times 8 = \underline{NOT(\overline{A_{in}} - 1) \times 2} = BnA \times 2$

$OP4 = L2nA \times 16 = BnA \times 4 = NOT(\overline{A_{in}} - 1) \times 4$

● $\Rightarrow OP1 + OP2 = NOT(\overline{A_{in}} - 1) \times 5$

$OP3 + OP4 = NOT(\overline{A_{in}} - 1) \times 6$

當然，不可以 $\overline{A_{in}}$ 表示

2's complement 為 $2^n - A_{in}$

~~$BnA \times 1 = (NOT(BnA \times 1)) + 1 = 2^{12} - A_{in}$~~

● $BnA \times 1 = BnA(8, 8, 8, 8:0) = NOT(2^{12} - A_{in} - 1) = L2nA \times 4$

~~$BnA \times 2 = BnA(9, 8, 8, 8:0) \& 0 = NOT(2^{11} - A_{in} - 1) \times 2 = L2nA \times 8$~~

$BnA \times 4 = BnA(7, 7, 7:0) \& 00 = NOT(2^{10} - A_{in} - 1) \times 4 = L2nA \times 16$

~~$L2nA \times 32 = BnA \& 000 = A_{in} \times 8$~~

$Brig\overset{E}{\cancel{htness}} = Brightness(7, 7, 7:0) = NOT(2^{10} - Brightness - 1)$

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2000/3/28

若 ARt = 11111 = 31

=> op1 + op2 = 2^{12} · Ain ~~...~~ /4

op3 + op4 = Ain × 8

~~=> B A × ARt = ...~~

=> BnA × ARt12 (11, 11, 11:2) = NOT (2^{14} - (2^{12}·Ain) - 1) / 4

= NOT (2^{14} - 2^{12} + Ain - 1) / 4

=> BnA × ARt = Ain × 8 + NOT (2^{14} - 2^{12} + Ain - 1) / 4

原始的 Contrast - Brightness 中

Brightness 定義 -128 ~ 127 與現在同

Contrast 為 0 ~ 255 /128 ∴ = 0 ~ 2 (255/128)

但和現在排法不同

昔 0 ~ 7F 80 -- FF
0 127 128 255
今 128 255 0 127 /128

sat_cos sin 0 127 -128 -1 / ~~128~~ 64

∴ contrast = sat_cos or sin + 128 / 128

or sat_cos or sin = contrast - 128 / 128

Ain × contrast { 00000000 ~ 01111111 10000000 ~ 11111111
~~0~~ 128 ~~127~~ 255 ~~128~~ 0 ~~255~~ 127
=> 設 contrast (?) 0 -> Ain 1 -> 0

or Ain × sat_cos or sin 00000000 ~ 01111111 10000000 ~ 11111111
0 127 -128 -1

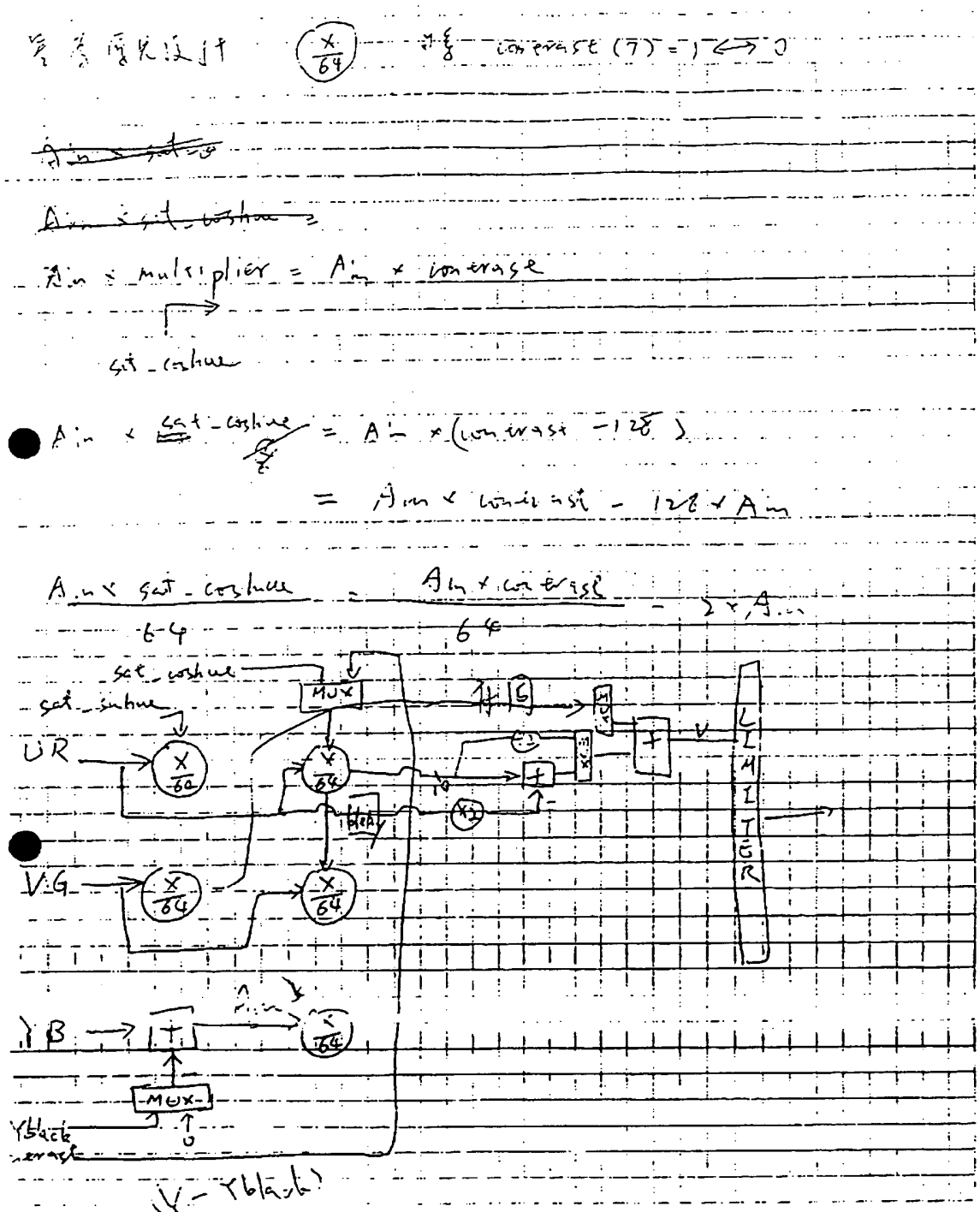
Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 20/3/78

17026

Subject: sat_cohue & cat_similar study



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2009/3/25

Subject: Bri-Inverse GS source code study

Ain d1 <= '0' & Ain ;
<8:0> <8> <7:0>

BmAx1 = ~~Ain~~ '0000' & Ain = Ain = L2mAx4

BmAxm1 = (NOT (BmAx1)) + 1 = 2^{12} - Ain

BmAx2 = Ain × 2 = L2mAx8

BmAx4 = Ain × 4 = L2mAx16

L2mAx32 = Ain × 8

VHDL code 中

process (sensitivity list)

↓

最好把所有變數應傳入之值列入

才會隨時間而隨時傳入

Y, U, V 有可能是負數

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

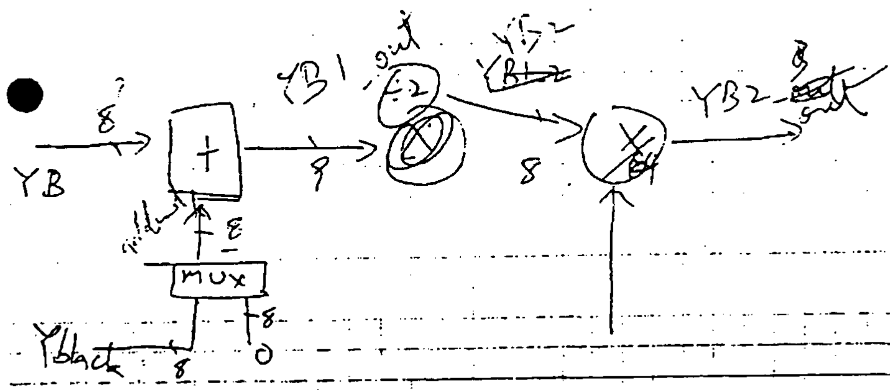
Date: 2000/3/28

ubject: 17028 Y black discuss

一、Y-black + Y 可能了 -128 + (-128) => -256.

二、把後面的 ÷2 移到前面

加法需增加 1 bit



⇓

会有一 1-bit 的失真

sclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2003/3/29

Subject:

Y、U、V、R、G、B 皆為正值

=> Y – Yblack 可能為負

Y 和 Yblack 皆為 8 bit

Y : 0~255 ; Yblack : -128~127

=> 第 8 個 bit (MSB) 表示不同 Y : 128~255
Yblack : -128~0-1

=> 各增加一個 bit

Y (7:0) -> 0 & Y(7:0)

Yblack (7:0) -> Yblack(7) & Yblack (7:0)

– (Yblack (8:0)) = NOT (Yblack (7, 7:0)) + 1

assume Yblack = 00000001

=> – Yblack (9bit) = 111111111

若 Y – Yblack 為負數，L1 之後 => Ain 為負

=> Y_out 為最大 ∴ Ain 做補數 (在 Multiply 中)

MUX 在後，若為正為 Y – Yblack 為負數

=> 轉為正數 => (x/64) -> 再轉為補數 + brightness;

Disclosed To And Understood By Me

Name Date

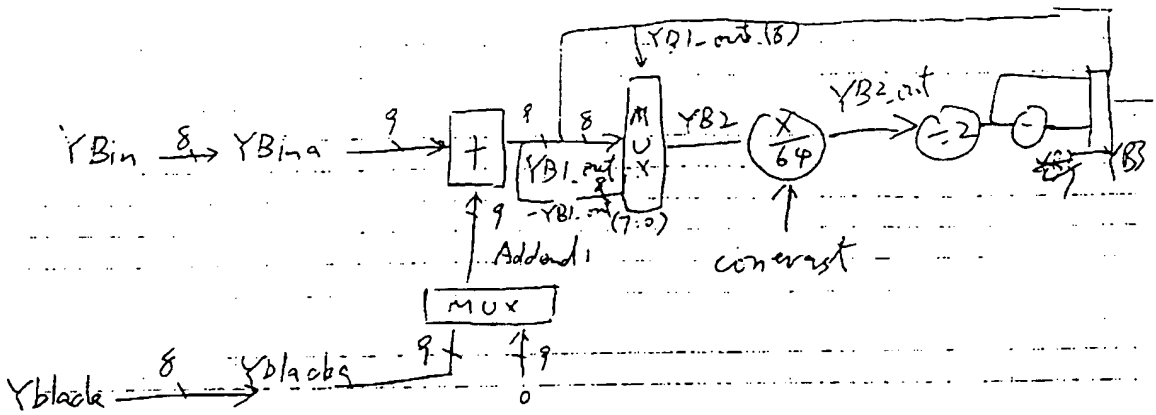
Name Date

Signature: blair

Date: 2000/3/20

17030

Subject:



or another way is

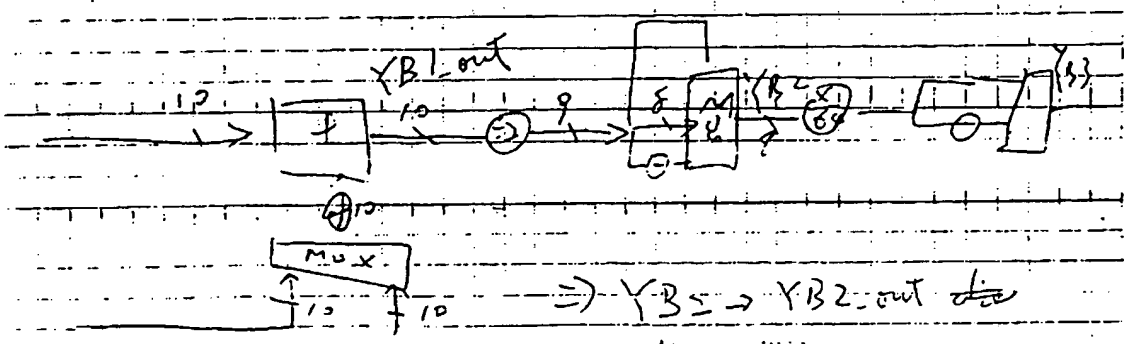
YBin : ~ 255 → 9 bit

YBlack : -255 ~ 127 → 9 bit

⇒ YBin - YBlack = -127 ~ (255+128)

= -127 ~ 383 → 9 bit

For YUV



⇒ YB = YB2_out + dissertation

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 00/3/30

Subject: _____ 17031

~~IF Lat YB1_out~~

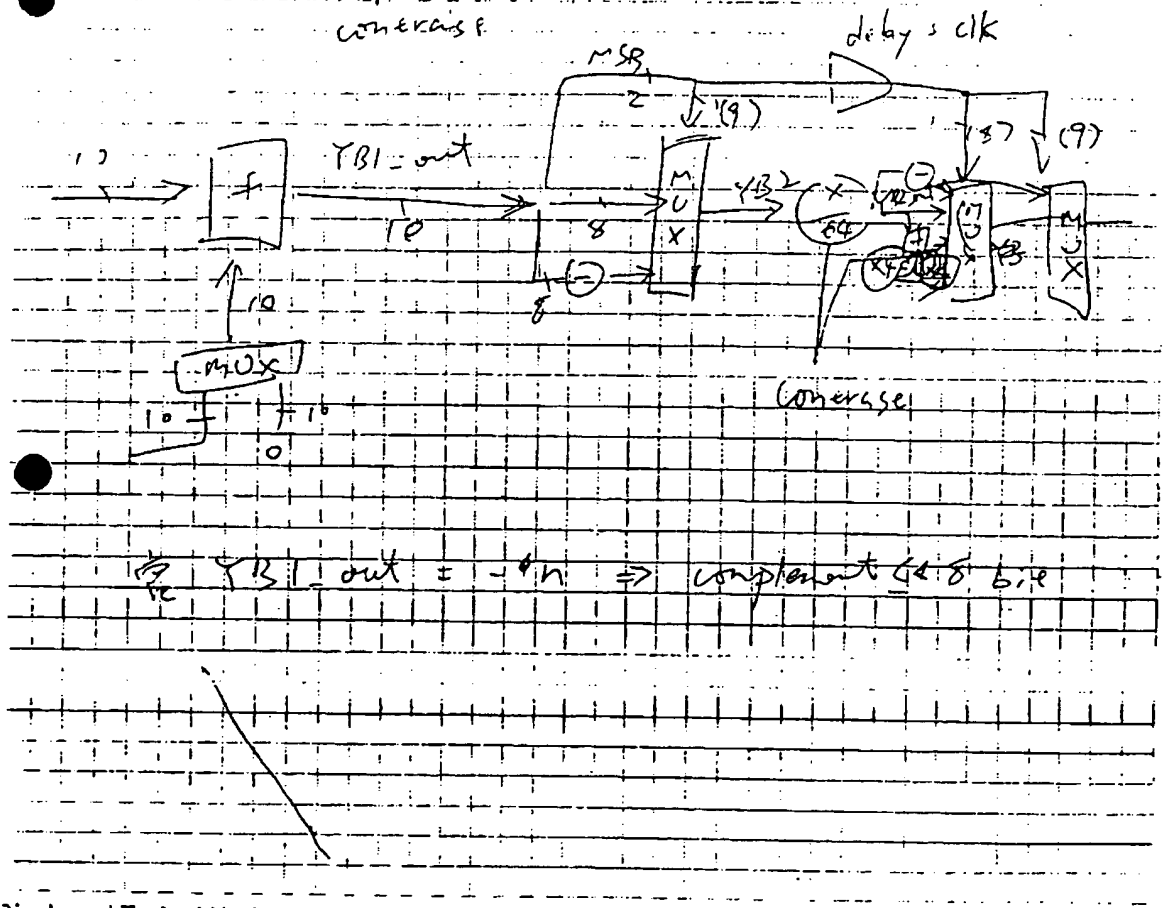
IF YB1_out ≥ 256 = ~~256+4~~

=> YB1_out = 256+n

= 100000000 + n

256 => (256) => converse + 4

converse



If YB1_out = -n => complement 4 & bit

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2003/3/30

Subject: 17032 YUV & RGB range

$Y_{out} = (Y - Y_{black}) \times contrast + brightness$

$U_{out} = U \times sat\_cos hue + V \times sat\_sin hue$

$V_{out} = V \times sat\_cos hue - U \times sat\_sin hue$

$RGB_{out} = RGB \times contrast + brightness$

input 값은 Y, R, G, B, U, V 0 ~ 255

- sat_sin hue : $-\frac{128}{64} \sim \frac{127}{64}$
- sat_cos hue : $\frac{-128}{64} \sim \frac{127}{64}$
- Yblack : -128 ~ 127
- contrast : 0 ~ 255/128
- brightness : -128 ~ 127

- $Y_{out} = (0-127) \times \frac{255}{128} - 128 \sim (255+128) \times \frac{255}{128} + 127$

$= -581 \sim 890$

$U_{out} = 255 \times \left(\frac{-128}{64}\right) + 255 \times \frac{-128}{64} \sim 255 \times \frac{127}{64} + 255 \times \frac{127}{64}$

$V_{out} = 1020 \sim 1012$

$V_{out} = 255 \times \frac{-128}{64} - 255 \times \frac{127}{64} \sim 255 \times \frac{127}{64} - 255 \times \frac{-128}{64}$

$= -1016 \sim 1016$

$RGB_{out} = 255 \times (0 \sim \frac{255}{128}) + (-128 \sim 127) = -128 \sim 639$

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2006/3/30

Subject: 17033

加强 brightness 的因素

Y_0 : -253 ~ 763 => 11bit

U_0 : ~~-1020 ~ 1012~~ => 10bit

V_0 : -1016 ~ 1016 => 10bit

RGB_0 : 0 ~ 508 -> 10bit

- inverse 0 — 7F 80 — FF (0, 127, 128, 255)

128/128, 255/128, 0/128, 127/128

↳ 7th bit transfer make the right value

Y_out 11bits

U_out 11bits

V_out 11bits

RGB_out 11bit

-128 ~ 127 -> 8 bit
-256 ~ 255 ~ 9 bit
-512 ~ 511 ~ 10 bit
-1024 ~ 1023 ~ 11 bit

testing : Y_out 的 order

負數 -> 0 >255 -> 255

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: [illegible]/3/30

TRUMPION CONFIDENTIAL BUSINESS INFORMATION, SUBJECT TO PROTECTIVE ORDER

TR000495

17034

Subject: Testing Environment

Y_out = (Y - Yblack) * contrast + brightness

[RGB_sel = 1]

1. 負數最大值 => Y = 0 => 00

Yblack = -127 => 7F

contrast = 255/127 => 7F

brightness = -128 => 80

2. 正數最大值 => Y = 255 => FF

Yblack = -128 => 80

Contrast ... => 7F

brightness = 127 => 7F

3. testing Y --- => Yblack = 0 => 00

contrast = 1 => 00

brightness = 0 => 00

(Bin = 0, => YB_out = 0

255 (FF) => YB_out = 255

4. testing Yblack => Y = 0 (00) =>

contrast = 1 (00)

brightness = 0 (00)

Yblack = 7F (+127) 00 (-0) 80 (-128)

Yout = 0 => FF => 0 => FF => 0 => 0 => 1 => 80

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2011/3/31

Subject:

17035

~~I. testing contrast~~ => Y = 128 (80) / 127 (7F) / 129 (81) Yblack = 0 (00) B = 0 (00)

C = 255/128 (7F) => Yout = 253 255 255

Y = 125 (7D) Yblack = 0 125 (7D) B = 0 (00)

C = 255/128 (7F) => Yout = 0 or random

Y = 125 (7d) Yblack = 7e

B = 0 . C =

1. testing contrast

① positive Y = ~~127 (7F) 128 (80)~~ 129 (81)

Yblack = 0 (00) B = 0 (00)

C = 255/128 (7F) 254 (7E) 253 (7D) 252 (7C) 127 (FF) 1 (81) 0 (80)

=> Yout = 258, 255, 254, 253, 127, 1, 0

② negative Y = 0 Yblack = 127 (7F) B = 127 (7F) ~~B = 0 (00)~~

C = 255/128 (7F) 128 (FF) 128 (00) 0 (80) 127 (FF) 126 (FE)

=> Yout = 0 ~~126 (7E)~~ 0 0, 127 (7F) 0, 1 (01)

Yout = FF -> FF -> FE -> FD -> 7F -> 01 -> 0

50

-> 0 -> 0 -> 7F -> 0 -> 1

50

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2002/3/31

17036

Subject:

6. testing brightness

(Y - Yblack) x contrast = 127

Y = 127 = (7F)

Yblack = 0 = (00)

contrast = (00) = 128

bright = -128 (80), -127 (81), -126 (82), 0 (00), 127 (7F)

=> Y_out = 0, 0, 1, 7F (127), 254 (FE)

∴ Total Y_out For equation line is

Y_out = 0 -> FF -> 0 -> FE -> 0 -> 0 -> 1 -> 80 ✓

-> FE -> FF -> FE -> FD -> 7F -> 01 -> 0 ✓

-> 0 -> 0 -> 7F -> 0 -> 1 ✓

-> 0 -> 0 -> 1 -> 7F -> FE ✓

YB-0- 0FE — 07E — 0 — 7FF — 0FE

waveform

=> YB1 -> YB-0 => Bri 延遲 delay 1 clk (simulation 3 [?])

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2000/3/28

Subject: RGB test

~~RGB × RGB~~

RGB-out = RGB × contrast + brightness

RGB_YUV_sel = 0

RGB 0~255

contrast 0~ 255/128

brightness -128~127

1. 最正值 -> RGB = 255 (FF) C = 255/128 (7F), B = 127 (7F)

=> OUT = FF

2. 最负值 = RGB × Contrast + brightness

= -128 (80)

{RGB = 2 AA, BB, CC 0
{contrast = 0 (80)

=> OUT = 0, 0

3. change RGB contrast = 128/128 (00) B = 0 (00)

RGB = 10, 11, FE, FF, 7F, 80

=> OUT = 0, 1, FE, FF, 7F, 80

4. change contrast

RGB = 129 (81) B = 0 Contrast = 0 (80), 127/128 (FF), 128/128 (00)

255/128 (7F), 254/128 (7E), 253/128 (7D)

OUT = 0, 127 (FF), 129 (81), FF, FE, FE

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2…/3 /10

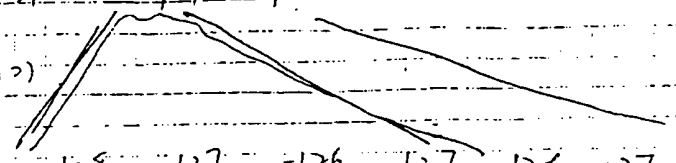
17038

Subject:

change brightness

RGB

RGBA CONTRAST = 127, 127, 81



Brightness = -128, -127, -126, 127, 126, 127

80 81 82 FF 7E 7F

OUT = 0, 0, 1, 2-4(FE), FF, FF

∴ Total RGB_out for ejection test is

RGB_out = FF → 0 → 0 → 0 → 1 → FE → FF → 7F → 80

→ 0 → 7F → 81 → FF → FE → FE

→ 0 → 0 → 1 → FE → FF → FE

sat_cosine → scos

sat_sine → ssin

U_out = U scos + V ssin

V_out = V scos - U ssin

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2007/5/24

Subject: UV test

④ RGB-YUV-sel → delay 7 times → next stage
U . V scos ssin 4 variable

1. U 的最大正數值 ⇒ U = V = 255 = FF

scos = 127/64 = scos (7F)

⇒ Uout = 255 , Vout = 0

2. U 的 最大負值 ⇒ U = V = 255 = FF

scos = 127/64 (7F) ssin = -2 (80)

U . V : 0 ~ 255

scos ssin = ssin 127/64 ⇒ V = 0, U = 0, U.V ≠ 0

1. V = 0
U = 255 (7F)
scos = -2 (80)

scos = -2 (80)

U_out : 1012 ~ -1012
V_out

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair
Date: 2001/3/20

1/04/13

Subject: UV test

U_out = U cos + V sin

V_out = V cos − U sin

1. variable : U 0~255

cos = $\frac{63}{64}$ (41)

V = 0

sin = − $\frac{63}{64}$ (BF)

U : 0, 255, 254 (FE), 253 (FD)

U_out : 0, FF, FE, FD

V_out : 0, FF, FE, FD

2. Variable : V 0~255

cos = $\frac{63}{64}$ (C1)

U : 0

sin : $\frac{63}{64}$ (41)

V : 0, 255, 254, 253

U, V_out : 0, FF, FE, FD

3. Variable : cos : $\frac{128}{64}$ ~ $\frac{127}{64}$

U : V : sin : 0 (20)

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 1/04/13

Subject: ____

3. variable : scos $-\frac{128}{64} \sim \frac{127}{64}$

ssin : $\frac{127}{64}$ (7F) for U & $-\frac{127}{64}$ (81) for V

~~ssin $\frac{127}{64}$ (7F)~~

~~U : 65 . 64~~

① ssin = $\frac{127}{64}$ (7F) V = 64 (40)

U = 65 (41)

~~U_out =~~

scos = $-\frac{128}{64}$ (80), $-\frac{126}{64}$ (82), $-\frac{125}{64}$ (83), 0, $\frac{126}{64}$ (7E), $\frac{127}{64}$ (7F)

U_out = 0, 0, 1, 127 (7F), 254, 255

V_out = 0, 0, 0, 0, 0, 0

② ssin = $-\frac{127}{64}$ (81) U = 64 (40)

V = 66 (42)

scos = $-\frac{128}{64}$ (80), $-\frac{124}{64}$ (84), $-\frac{123}{64}$, 0, $\frac{124}{64}$ 7C, $\frac{125}{64}$ 7D, $\frac{127}{64}$ 7F

U = 0, 0, 0, 0, 0, 0, 0

V_out = 0, 0, 1, 127, 254, 255, 255

~~4. ...~~

4. variable : ssin

Disclosed To And Understood By Me

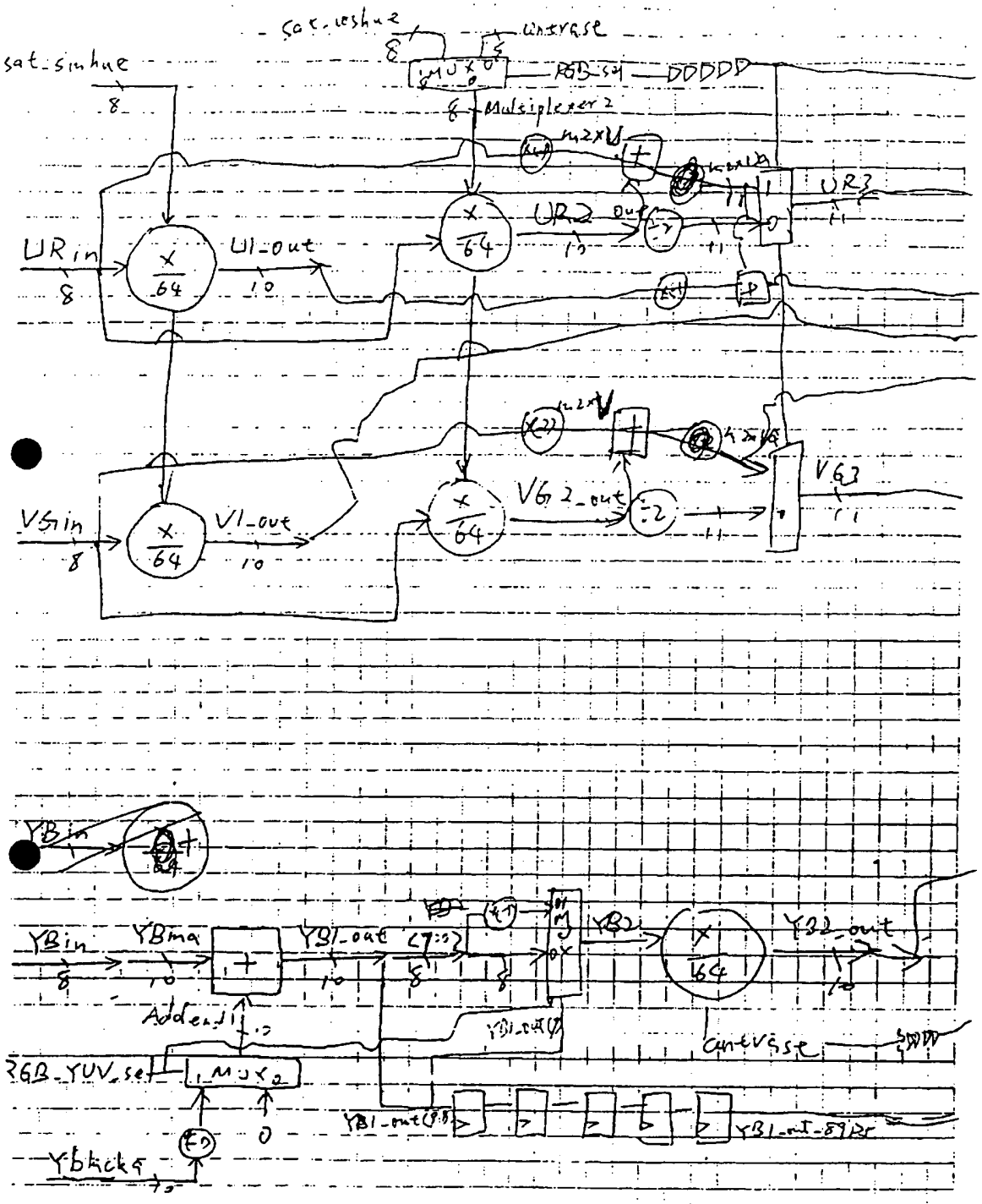
Name	Date
Name	Date

Signature: blair

Date: 2000/4/7

17042

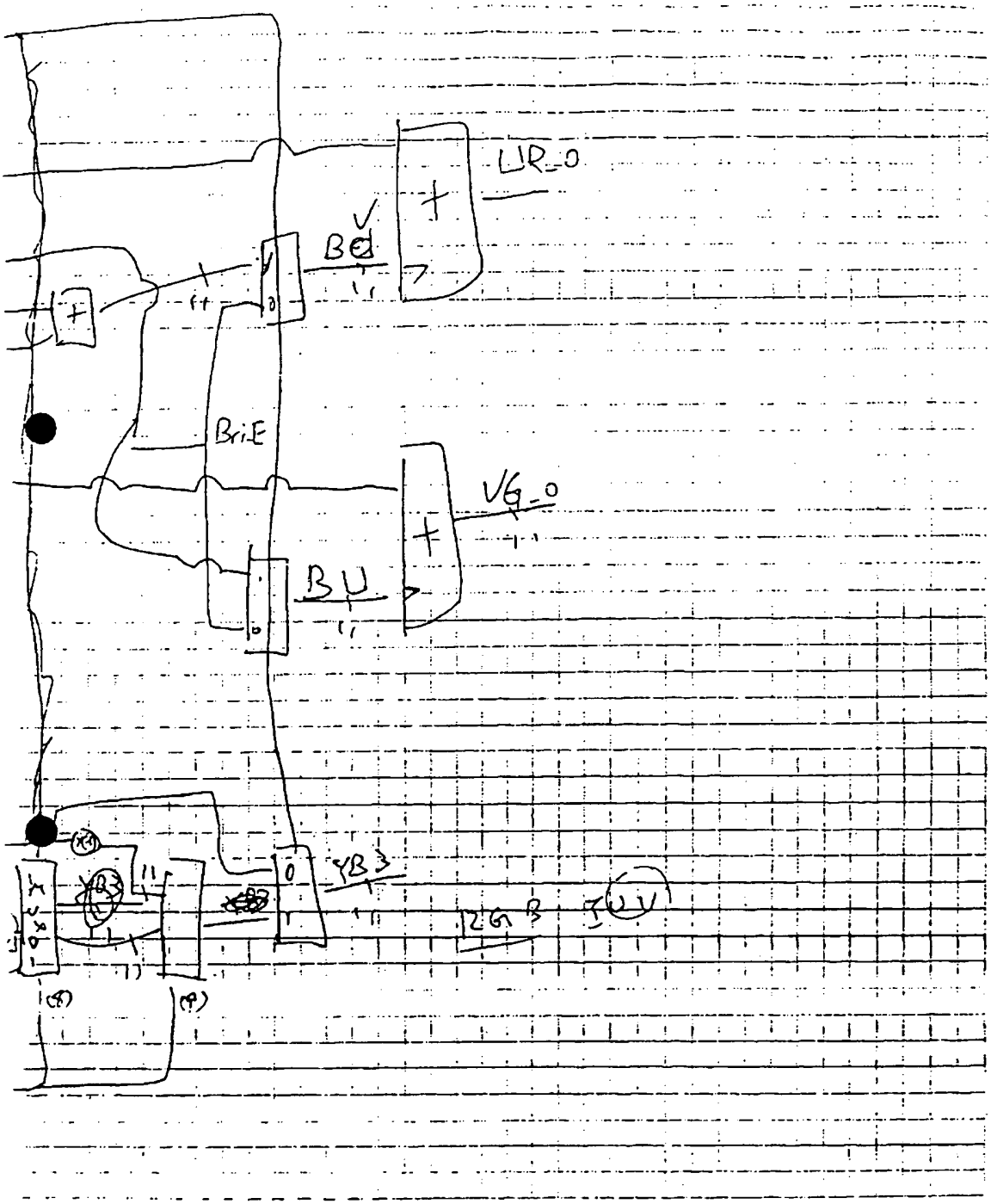
Subject: YUV_RGB flow



sclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 00/4/3

Subject: _____ 17043

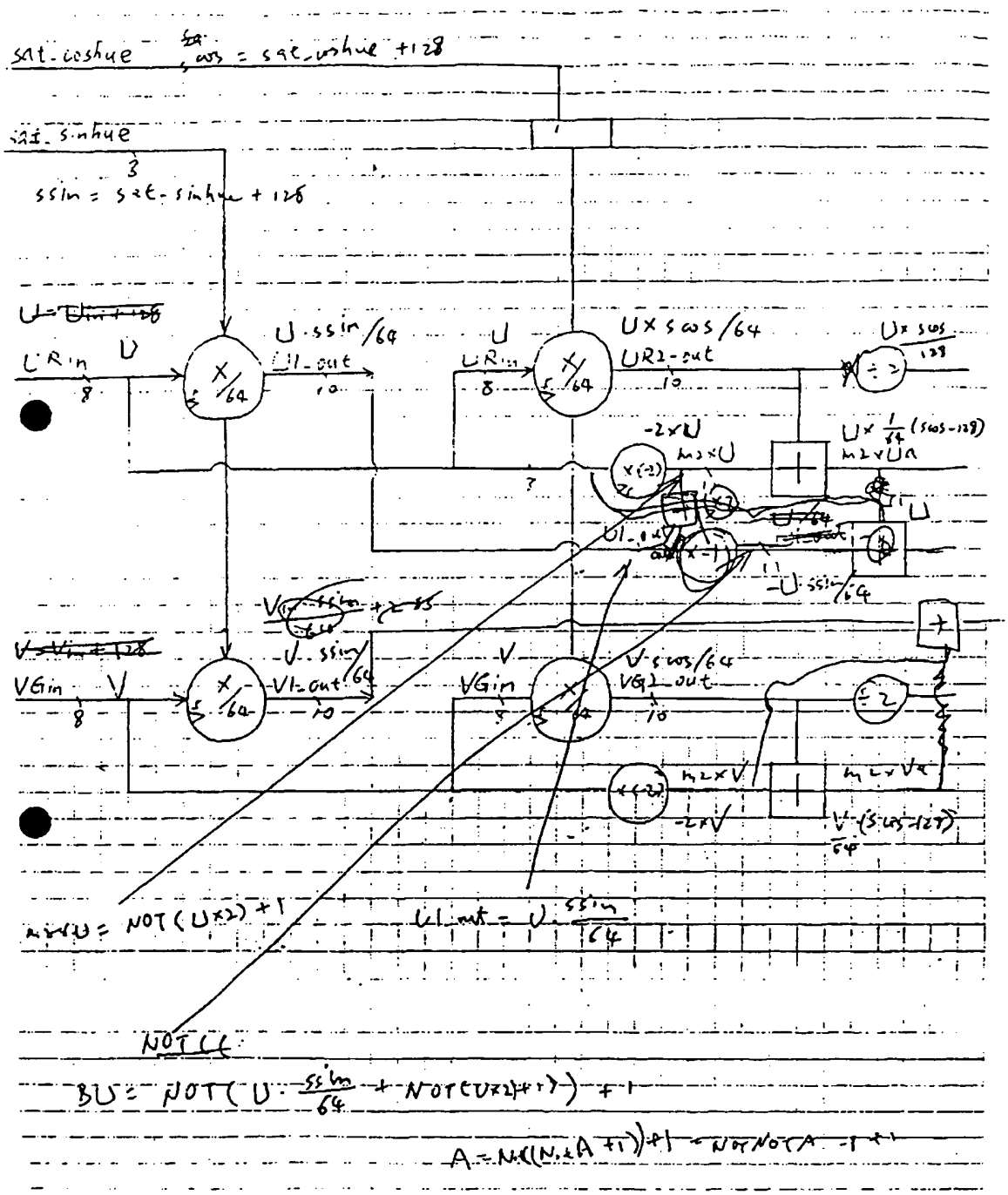


Disclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 2000/8/3

17044

Subject: [illegible] UV flow

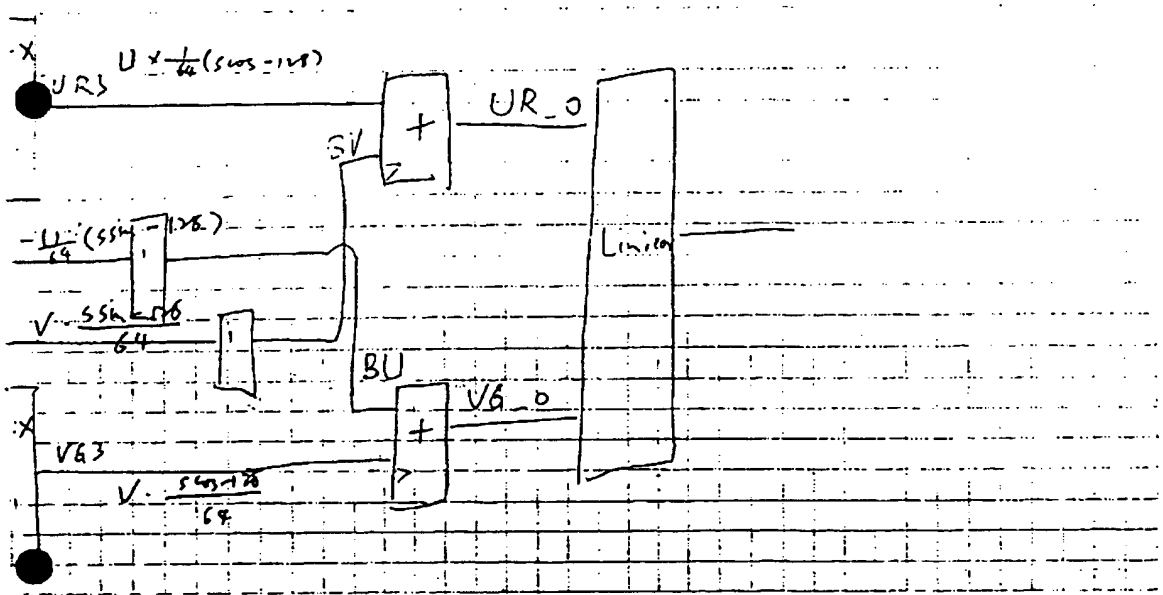


Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2020/4/5

Subject: _____

17045



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2003/4/5

17046

Subject: U ?) V test

4. variable : ssin

scos = + 127/64 (81) 7F   8 127/64 (9F) 81

① scos = + 127/64 (81) (7F)

U = 64 (40)   V = 66 (42)

ssin = -128/64 (80), -123/64 (85), -122/64 (86), 0, 124/64 (0C), 125/64 (7D), 127/64 (7F)

U = 0, 0, 1, 127, FE, FF, FF

V = 0 - 0 0 0 0 0 0

=> U_out = 0 0 1 127 254 ...

V_out = 127/64 × 64/66 ... ssin × 64/... = 127/64 × 66 ... ssin × 64

②③ ... => 0 ... 7F,

FF, FD, FC, 82, ..., 6, 5, 3

③ 最大最小值

U=255, V=255, scos = 127/64 (7F) = ssin

=> U_out FE V_out 0

ssin = -128/64 (80)

=> U_out 0, V_out FF

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2003/?/5

17047

Subject:

U = ~~25~~ = V

5 cos = $\frac{128}{64}$ (80) = 5 sin

=> U ≠ 0 V ≠ 0

5 sin = $\frac{127}{64}$ (7F)

=> V_out = 0 = V_out

4. ② 5 cos = + $\frac{127}{64}$ 7F ~~(5F)~~ U = 66 (42) V = 64 (40)

=> V_out = ~~0 64 15 / 127~~ − ~~64~~ 66. 5 sin / 127

=> 5 sin = $\frac{127}{64}$ (7F), $\frac{124}{64}$ (7C), $\frac{123}{64}$ (7B), 0 ~~$\frac{123}{64}$ (7B)~~ (85), $\frac{124}{64}$ (84), $\frac{127}{64}$ (80)

V_out = 0, 0, 1, 7F, 253 (FD), FE, FF

U_out = 127 × $\frac{66}{64}$ + 64 · 5 sin

= 130 + 64 · 5 sin

~~= 254, ..., FA, 7F~~

~~(FC) (FB)~~

= FF, FE, FC, 82, 7, 6, 3

① For 4 ①

U_out = 0, 0, 1, 7F, FE, FE, FF, 0 0 0 0 0 0 0

V_out = 0 0 0 0 0 0 0 0 0 1 7F FC FF FF

Disclosed To And Understood By Me

Name Date

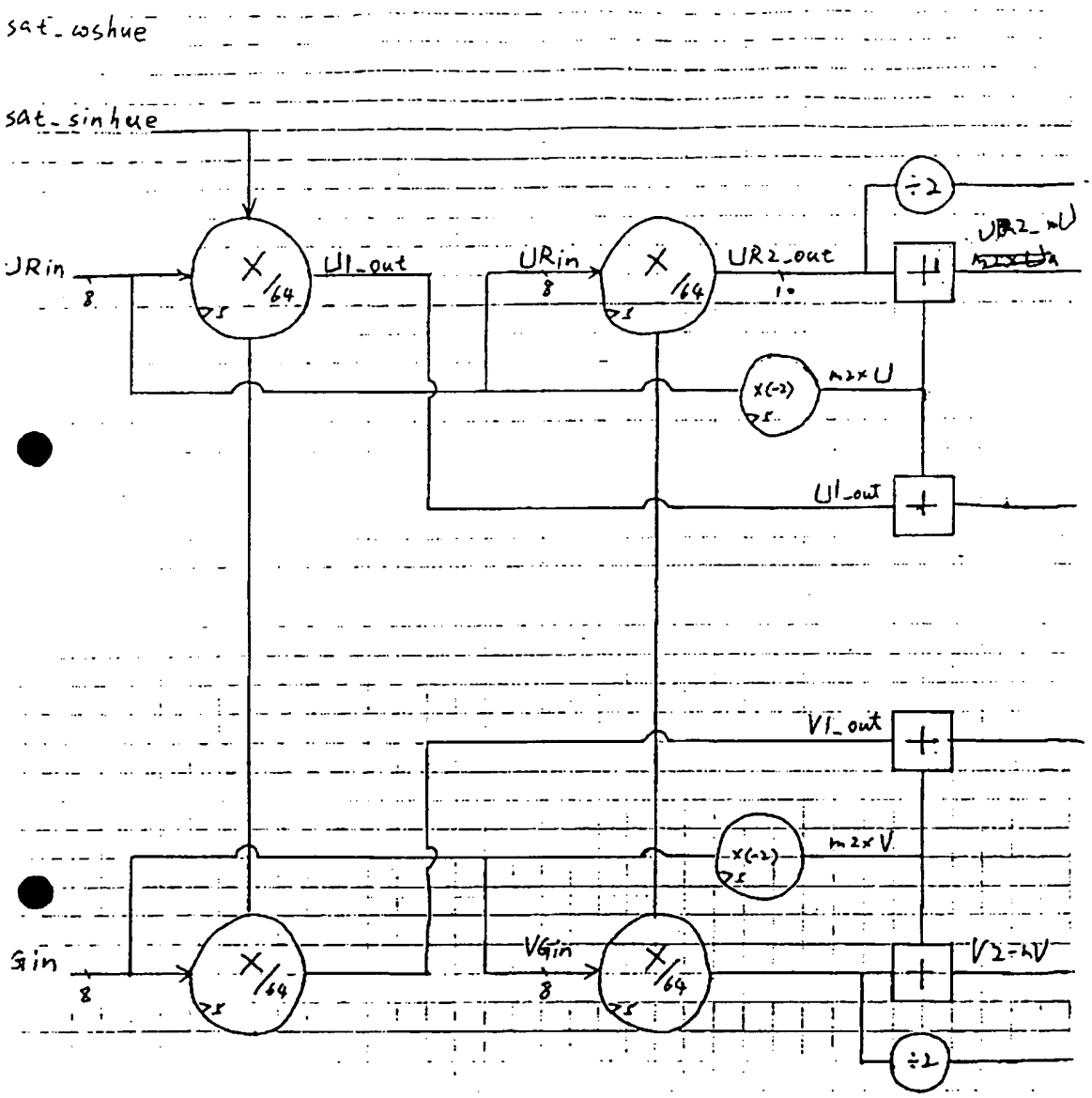
Name Date

Signature: b/o'r

Date: 2000/4/5

17048

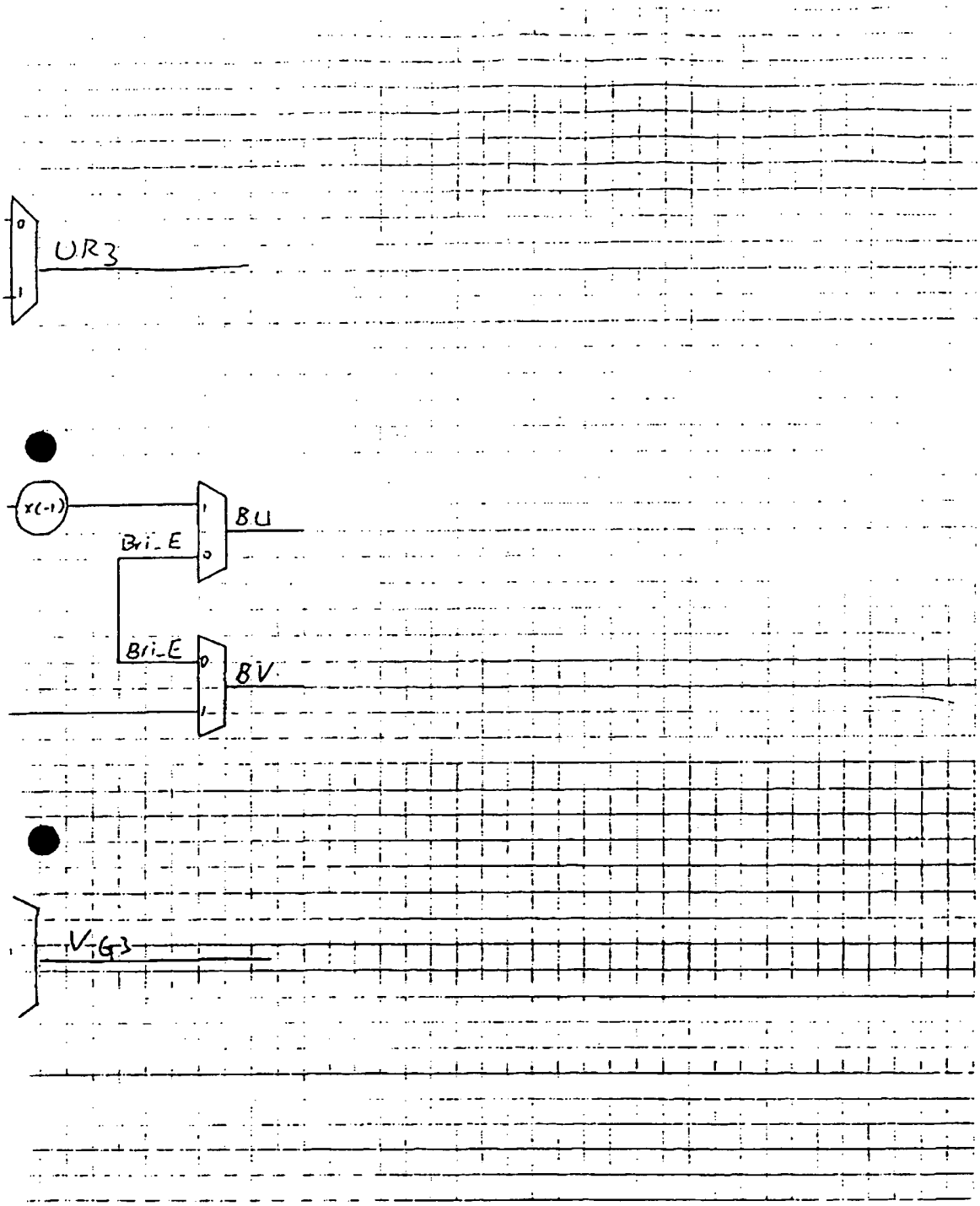
Subject: RGB - YUV Flow



Disclosed To And Understood By Me

Name	Date	Signature:	
Name	Date	Date:	2000/4/15

Subject: _____



Disclosed To And Understood By Me

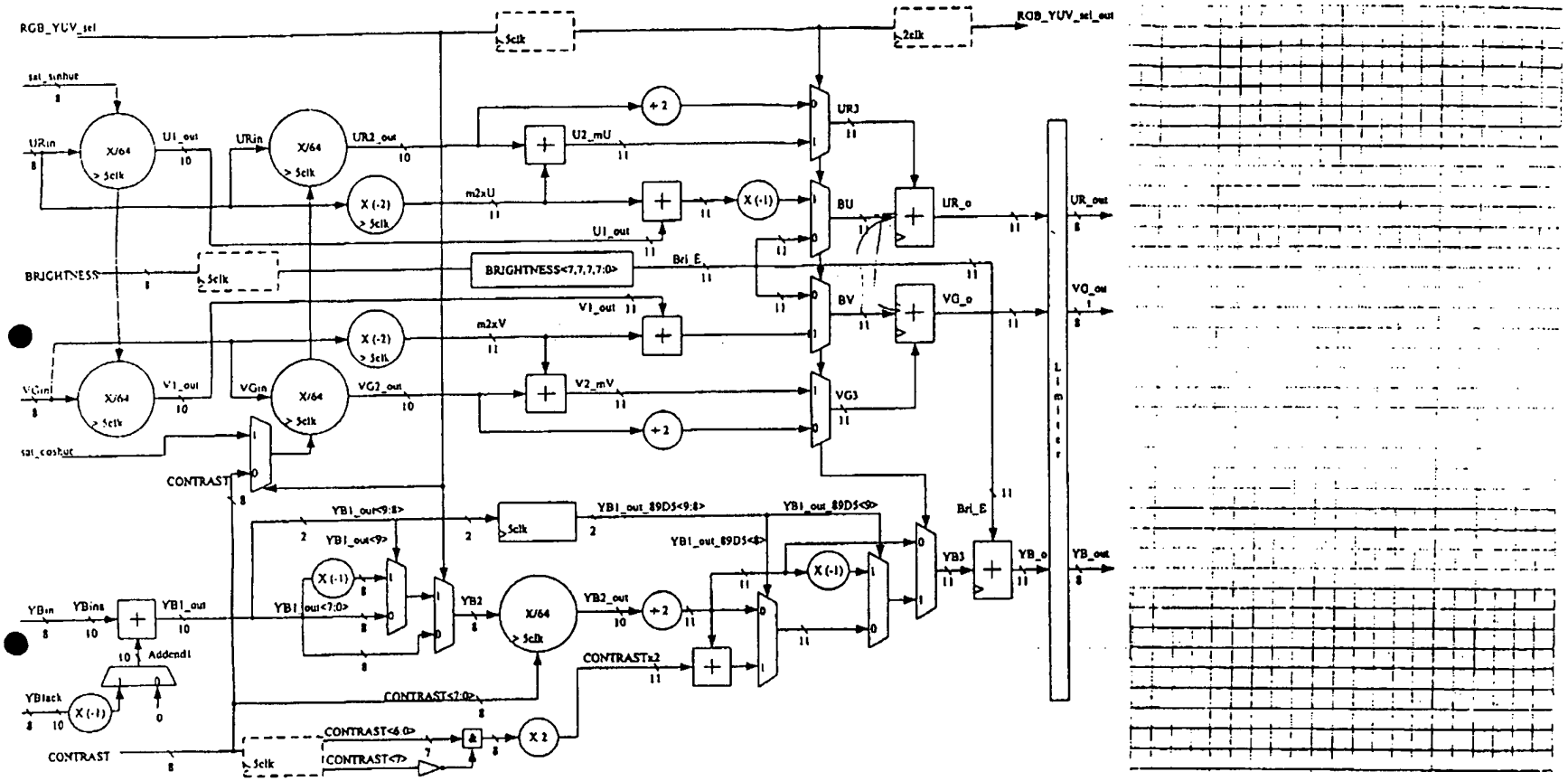
Name	Date	Signature: Llair
Name	Date	Date: nov 9 15

17050

Subject: _____

17051

Subject: _____



Disclosed To And Understood By Me

Name _____ Date _____

Name _____ Date _____

Signature: blair

Date: 2000/4/6

Disclosed To And Understood By Me

Name _____ Date _____

Name _____ Date _____

Signature: blair

Date: 2000/4/6

11002

Subject: Multiplier study

目前的架構為正數×正數，但可能可更改為

正數×(正負數)

↑ Multiplier (原 CONTRAST)　　↑ Ain

原架構中已考慮到 Multiplier (CONTRAST) 為 clk delay 的 issue。若 consider gate count，在正×正情形下，

● 可把 CONTRAST 的 delay 拿掉

若要改為(正負)×(正)，則在原架構中需要

增加位數之 MSB 並填為原數之 MSB 值

例如：B = PN×4 = PN(7) & PN(7) & PN(7:0)

試看看。

± A → PN　⇒ 補數時 ~~PN = NOT(A) + 1~~ ~~A = PN~~

● B → P　　A = PN ⇒ +A = ~~NOT~~ NOT(PN) + 1

± S → OUTPUT　　S = OUT ⇒ +S = ~~OUT~~ NOT(OUT) + 1

根據 simulation 的結果，it ~~st~~ seems o.k.

Disclosed To And Understood By Me

Name　　Date

Name　　Date

Signature: blair

Date: 2000/4/7

Subject:

P = 23 , PN = -1 => OP1, 2, 3, 4 的需用 ...

=> Xdiv64 = 3FE = 11,1111,1... = -1

P = 23 (17) PN = ~~40~~ -64 (C0) => 3E9 (= 23 DEC = 17 HEX)

~~P=~~ P = 23 => P = 10111

P = 1111 0111 => 247 (F7) PN = -64 (C0) => 309

Xdiv64 = 309 => F7 的補數

~~CPN_d3 = PN × P<7:6> + PN<7> & "0000000000" × P<7:6>~~ (9)

~~+ PN<7> & "0000000000"~~ (10)

CPN_d2 := PN<7,7:0> × 2P<7> + PN<7,7,7:0> × P<6>

= PN<7:0> × P<7:6> + PN<7> & "000000000" × 2P<7>

+ PN<7> × 11 × "000000000" × P<6>

= PN × P<7:6> + PN<7> × 512 P<7> + PN<7> × 11 × 256 P<6>

↳ if PN<7> = 0 => 0

PN<7> = 1 => 1 × 512 P<7> + 3 × 256 P<6>

=> 2 × 256 P<7> + 3 × 256 P<6>

=> PN<7> × 256 P<7:6> + PN<7> × 256 P<6> × 2

=> PN<7> × 256 × (P<7:6> + 2P<6>)

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2000/4/10

17054

Subject:

$pN \times 2P<7:6> + PN<7> \times 512 \times (P<7:6> + 2P<6>)$

$+ PN<7> \times 7 \times 256 \times P<5>$ ~~$\times PN<5> \times P6$~~

$+ pN \times p<5>$

$= PN \times P<7:6:5> + PN<7> \times 256 \times [2P<7:6> + 4P<6> + 7P<5>]$

$= PN \times P<7:6:5> + PN<7> \times 256 \times [P<7:5> + 3P<6:5> + 4P<5>]$

$[4P<7> + 4(P<6> + 7P<5>]$

● ~~$23 \times (-128) = H17 \times H80$~~

~~= 680 = 0110, 1000, 0000~~ = ~~1117~~ 的补数

2578

~~$23 \times (-128) = 017 \times F80$~~ => 16480 (HEX)

~~补数 => B4F11 = B80~~

= ~~1011, 1000, 0000~~

=> ~~FE17F 的补数~~ = ~~2944~~

● ~~$23 \times (-128)$~~ = 17 × 80 = 680 (HEX) = 0110, 1000, 0000

~~(1477)~~ = 180 的补数 = 384

$23 \times (-128) = 17 \times 80$ = 1,011 × 1000 0000

= 1011, 1000, 0010

= B80 = 480 的补数 = ~~1408~~ 1152

Disclosed To And Understood By Me

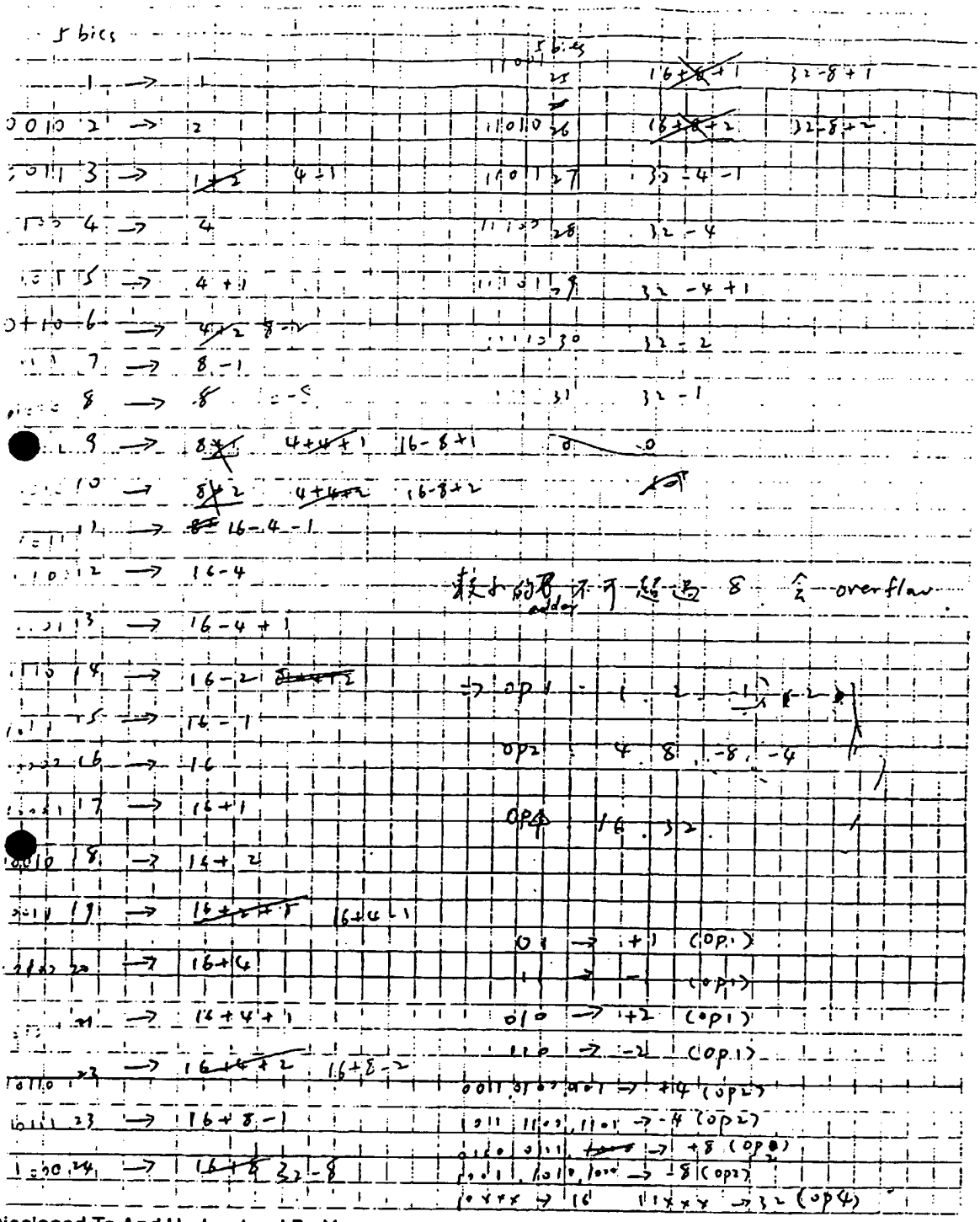
Name Date Signature: 3/c1v

Name Date Date: 200/ 4/2

 TR000515

Subject: 9 bits × 5 bits MUX

17055



Disclosed To And Understood By Me

Name Date

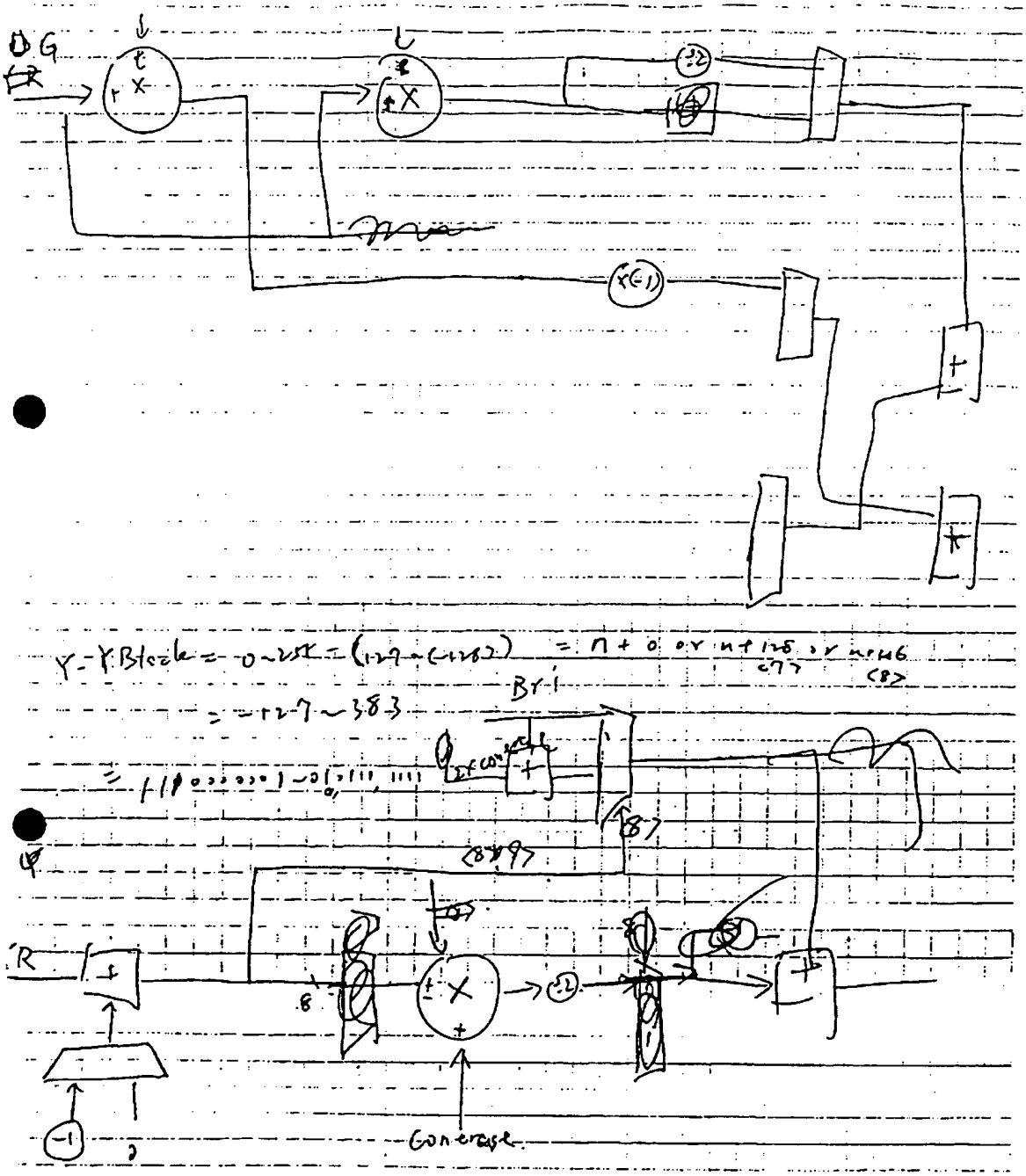
Name Date

Signature: blair

Date: 2000/4/10

17056 YUV - RGB

Subject: Converse under new designed PNXP/64



isclosed To And Understood By Me

Name Date

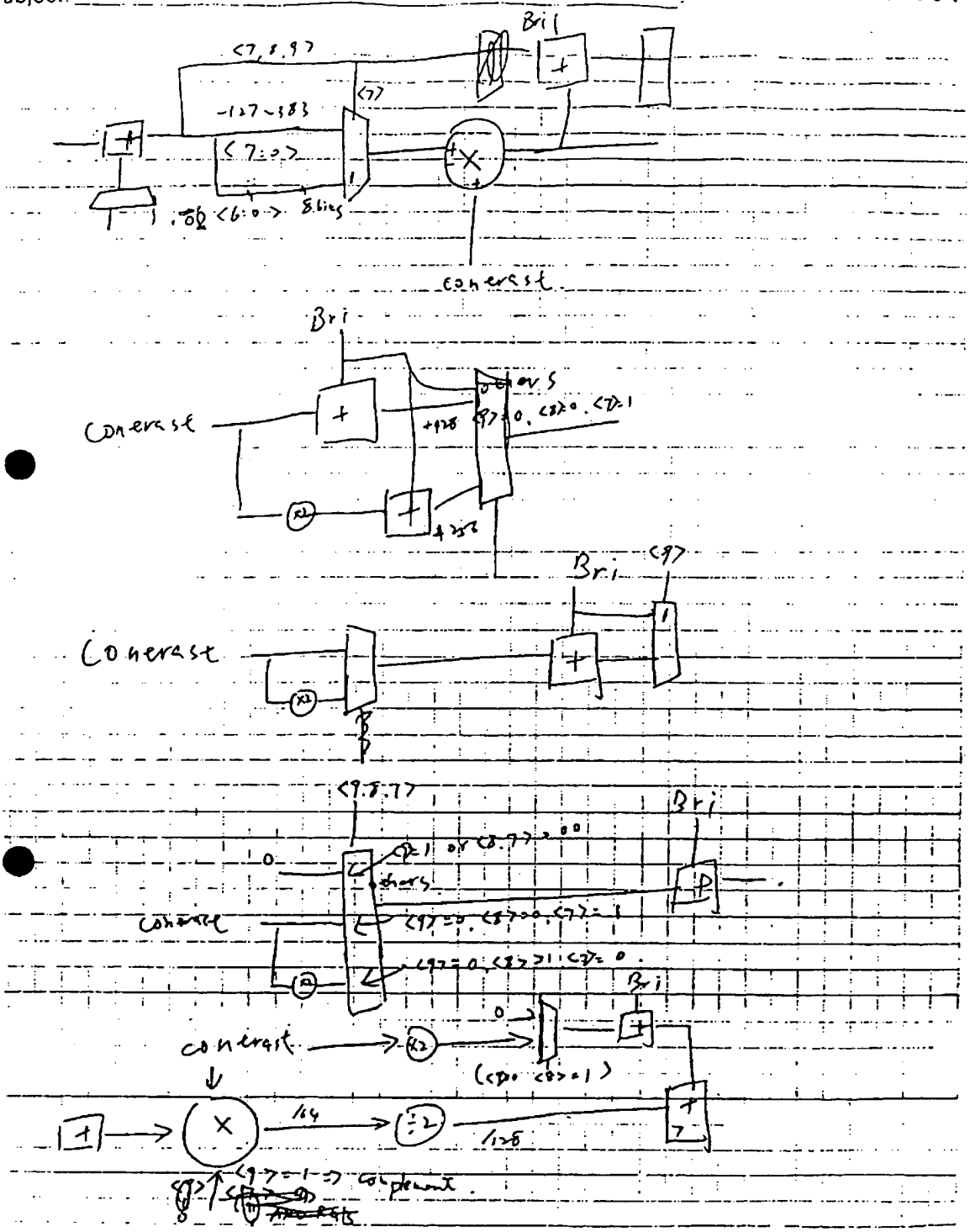
Name Date

Signature: blair

Date: 2000/4/11

Subject: _____

17057



isclosed To And Understood By Me

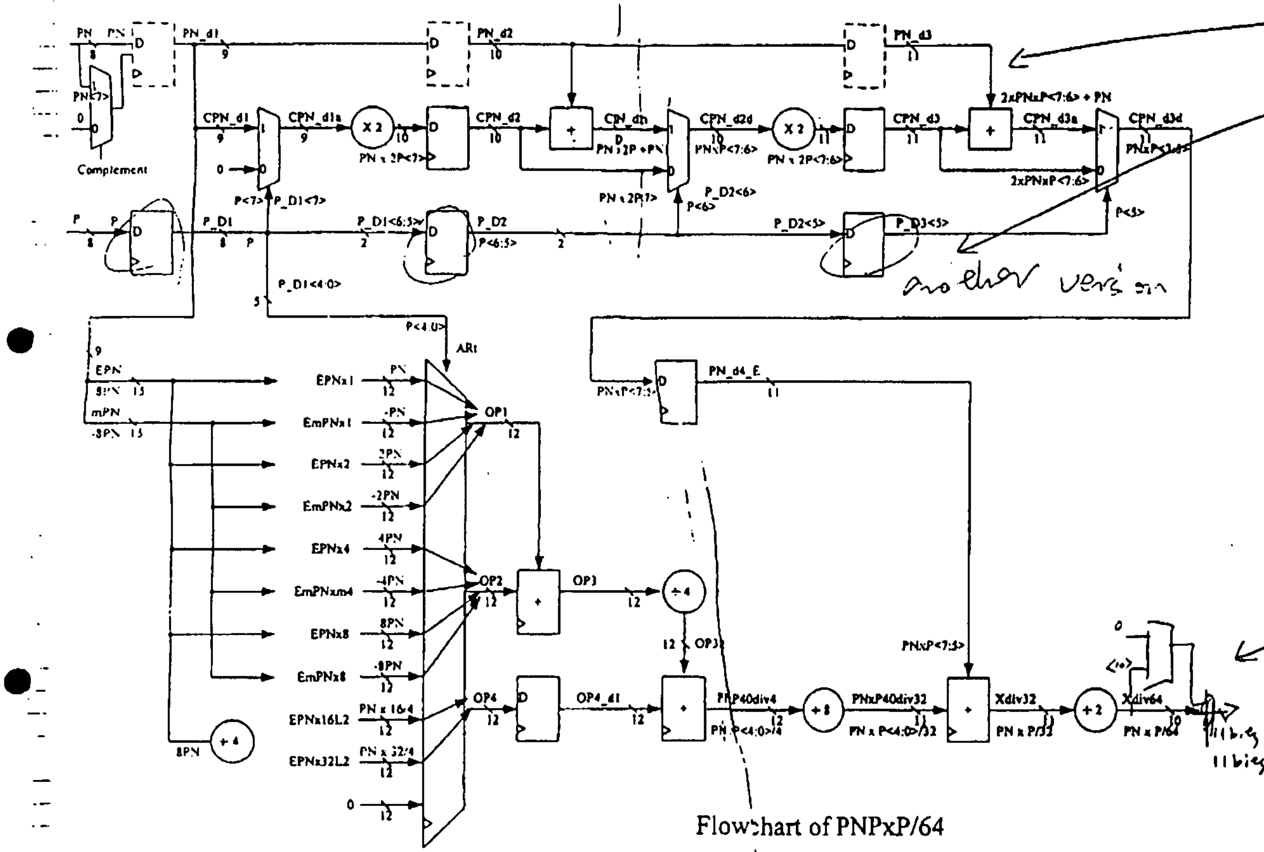
Name	Date	Signature: Blair
Name	Date	Date: 2003/4/12

17058

Subject: Multiplier improve

Subject:

17059



Flowchart of PNPxP/64

For circuit area issue we can remove DFF of PN or P if the input ones is not at often vary with time.

Thus reduce D-FF.

for PN inputs reduce 10 D-FF,

for P inputs reduce, 11 DFF

we add ~~this~~ a MSB to identify whether the result is positive or negative

If the MUX at the end is replaced in functionally at the middle of the design, will therefore adding a few D-FFs, so it is a easier and cheaper way for adding mux at the end.

Disclosed To And Understood By Me

Name Date

Name Date

Signature: b/sir

Date: 2003/4/12

Disclosed To And Understood By Me

Name Date

Name Date

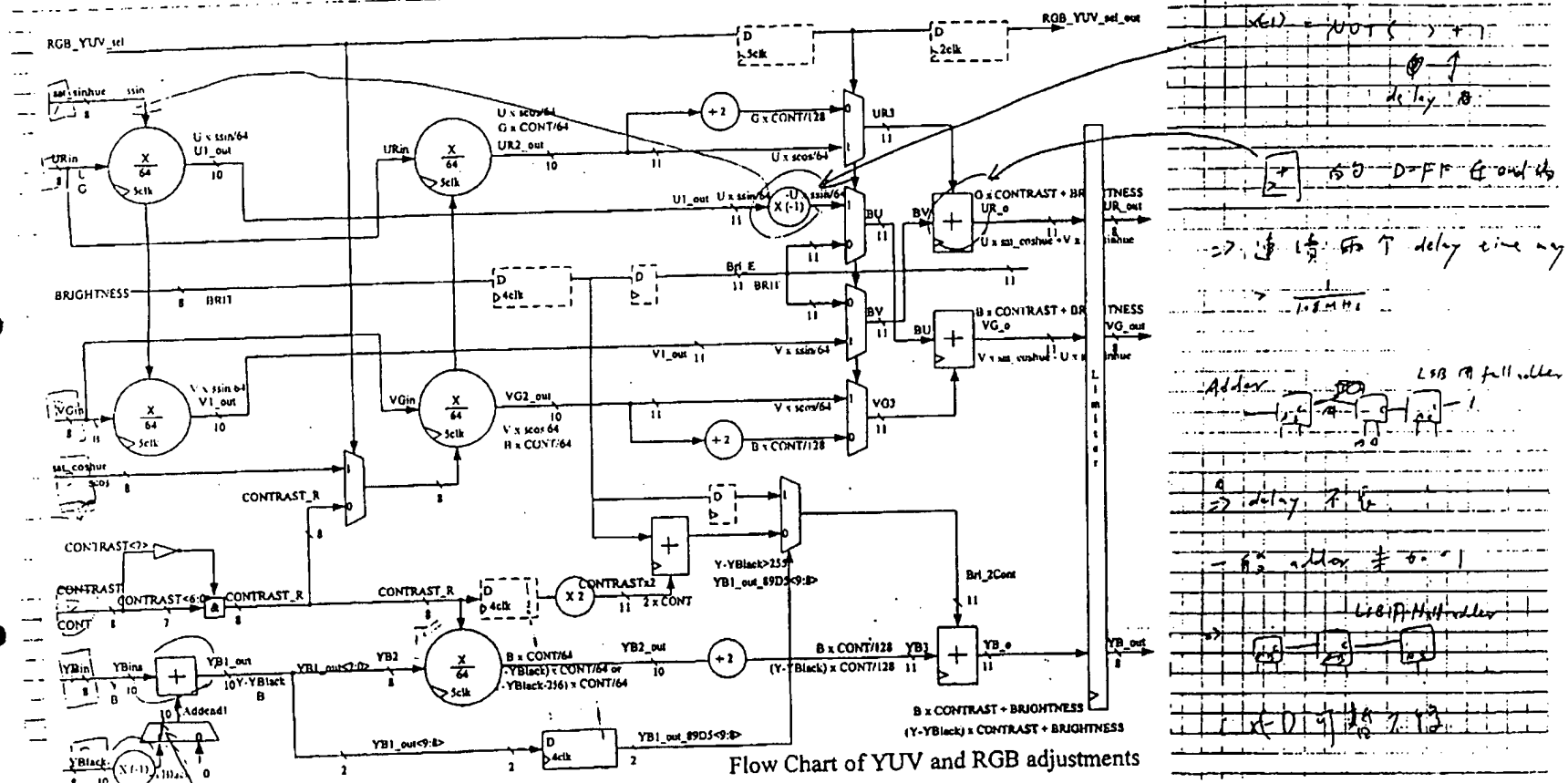
Signature: b/sir

Date: 2003/4/12

Subject: 17060 Delay issue of YUV adjustments

Subject:

17061



Flow Chart of YUV and RGB adjustments

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2020/4/12

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2020/4/12

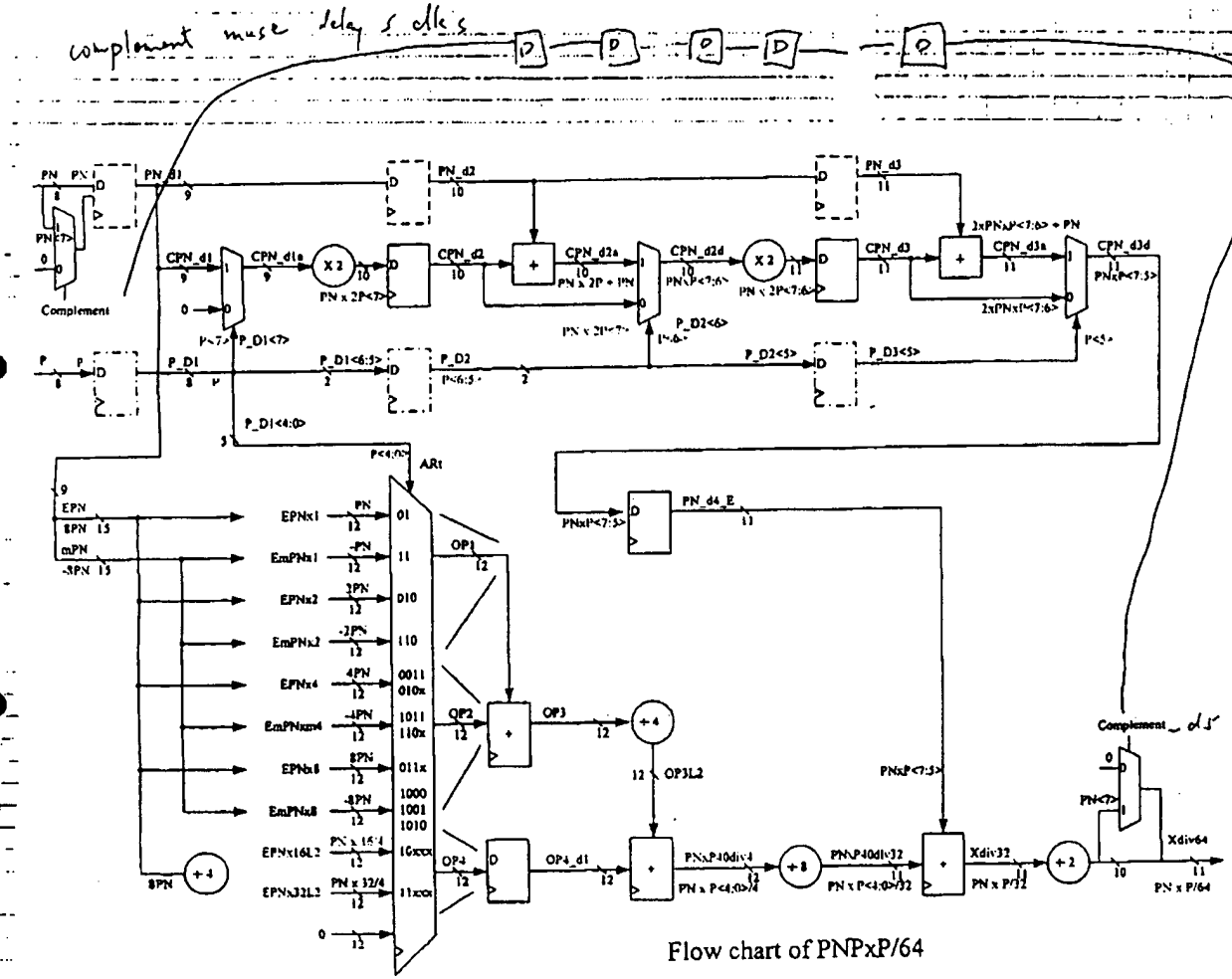
TRUMPION CONFIDENTIAL BUSINESS INFORMATION, SUBJECT TO PROTECTIVE ORDER

TR000520

Subject: 17062 complement signal in new designed multiplier

Subject: 17063

complement must delay 5 clks



Flow chart of PNPxP/64

in order to match the output of the complement controlled mux.

By the way PFF can be removed if complement is ~~rarely~~ hardly vary with time.

TR0000521

TRUMPION CONFIDENTIAL BUSINESS INFORMATION.
SUBJECT TO PROTECTIVE ORDER

Disclosed To And Understood By Me

Name Date

Name Date

Signature: [signature]

Date: 2008/4/13

Disclosed To And Understood By Me

Name Date

Name Date

Signature: [signature]

Date: 2008/4/13

17004

Subject: Testing RGB code

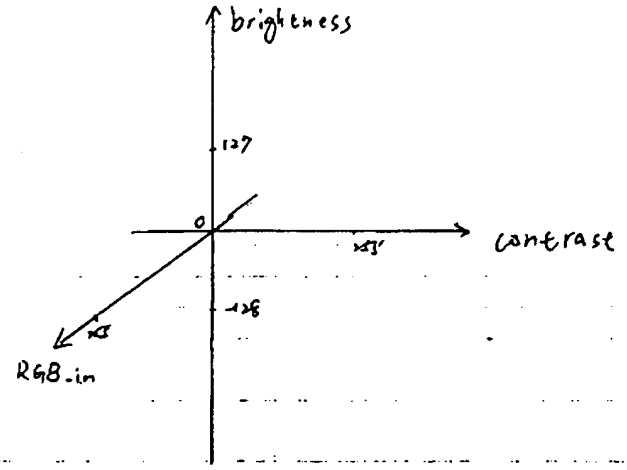
RGB_out = RGB_in × contrast + brightness

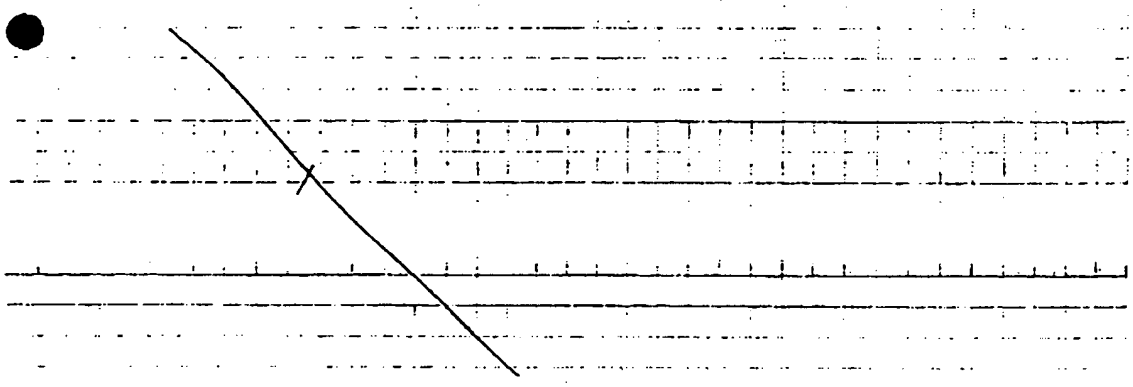
RGB_in : 0 (00) ~ 255 (FF)

RGB_out : 0 (00) ~ 255 (FF)

contrast : 0 (80) ~ $\frac{127}{128}$ (FF). $\frac{128}{128}$ (00) ~ $\frac{255}{128}$ (7F)

brightness : -128 (80) ~ -1 (FF). 0 (00) ~ 127 (7F)





sclosed To And Understood By Me

Name	Date
Name	Date

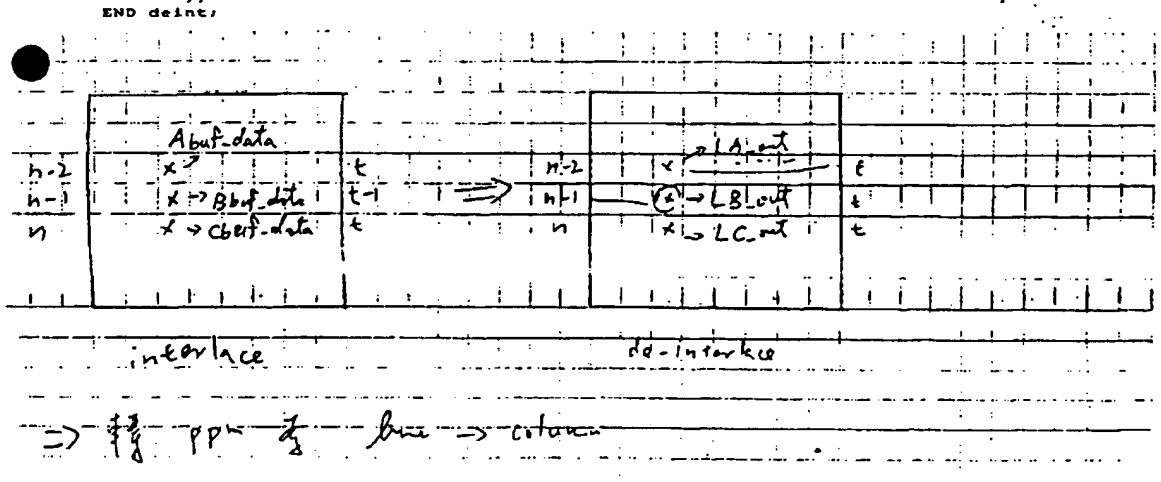
Signature: blair

Date: 2000/9/14

Subject: de-interlace testing

17005

```
File: D:\projects\Scaler\Oslo\deinterlace\deint.vhdl  2000/3/30, 11:37:56AM

LIBRARY ieee;

USE ieee.std_logic_1164.all;
USE ieee.std_logic_unsigned.all;

--      Gene Chuang, Mar 30, 2000, trumpion us+
--      name: deint.vhdl (deinterlacing circuit for video)

--      input: Abuf data (n-2 Frame t)          output:  LA_out (n-2 Frame t)
--             Bbuf data (n-1 Frame t-1)                 LB_out (n-1 Frame t)
--             Cbuf data (n   Frame t)                   LC_out (n   Frame t)
--
--      The deinterlacing ckt genesrates the 3 lines (n-2, n-1, n) for scaler
--
--      There are 3x3 pixels for motion detection and I to Progressive
conversion.
--      namely,
--      p1 = RGB(pA,delay1); p2 = RGB(pA,delay2); p3 = RGB(pA,delay3);
--                           p4 = RGB(pB,delay2);
--      p4 = RGB(pC,delay1); p5 = RGB(pC,delay2); p6 = RGB(pC,delay3);
--
--
--      (Diff81+Diff86 && Diff82+Diff85 && Diff83+Diff84 > THDmotion) means
Motion

--      Motion: if (Diff82+Diff85 < Diff81+Diff86 && Diff82+Diff85 <
Diff83+Diff84)
--                then the result is the averaged of pixel2 and 5.
--      Stationary: the result is from the previous field (F t-1).
--
--      Since PH_active (or swap_hs), powerdown, and PV_active are not used, to
save
--      power, the input is set to zero during display disable period.


ENTITY deint IS
  PORT (
        rstn            : in  std_logic;
        powerdown       : in  std_logic;
        clk             : in  std_logic;
        pv_active       : in  std_logic;
        swap_hs         : in  std_logic;
        abuf_data       : in std_logic_vector(23 downto 0);
        bbuf_data       : in std_logic_vector(23 downto 0);
        cbuf_data       : in std_logic_vector(23 downto 0);
        THDmotion       : in std_logic_vector(7 downto 0);
        LA_out          : out std_logic_vector(23 downto 0);
        LB_out          : out std_logic_vector(23 downto 0);
        LC_out          : out std_logic_vector(23 downto 0)
       );
END deint;
```

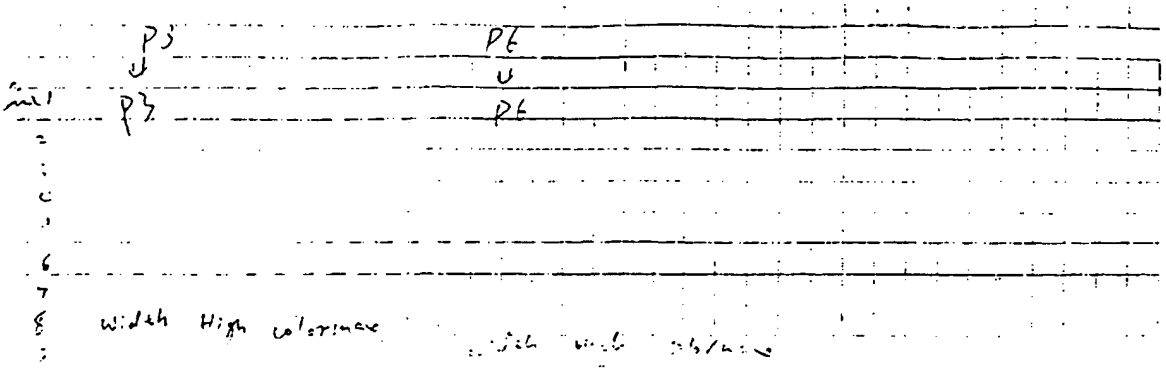


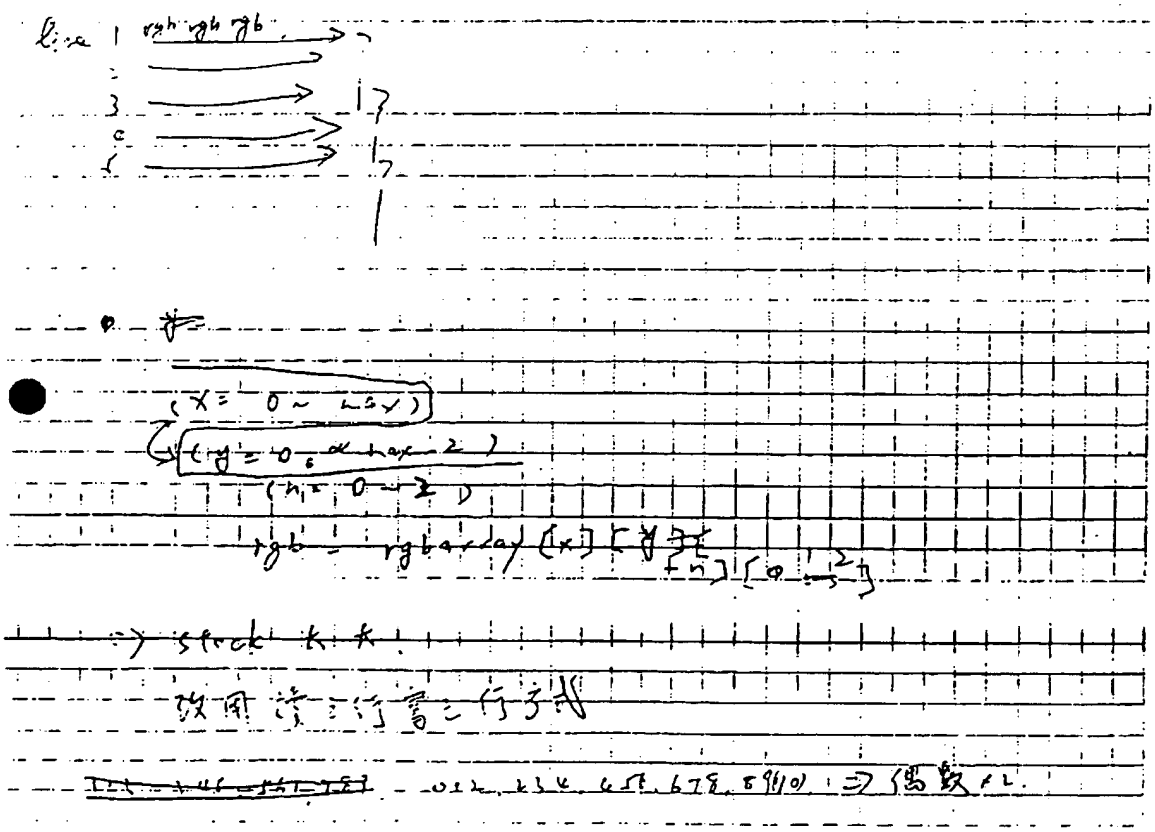
Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 4/5

Subject: 170CG PPm to RGB code study

PPM code format 가 P3 P6





Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2022/4/16

Subject: Modelsim lib compile method

17067

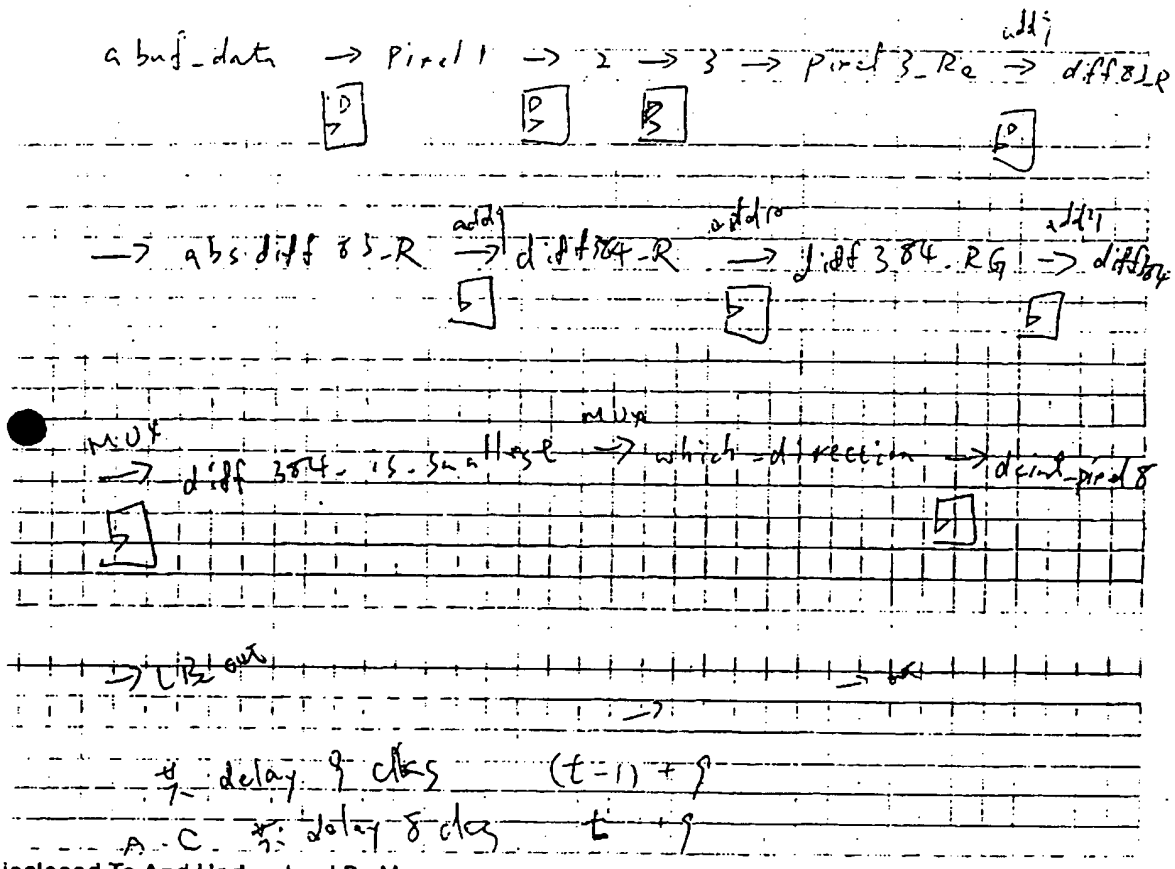
modelsim

vlib oslo_lib

vcom --work oslo_lib

modelsim.ini 에 lib 경로 add

- deint delay clock



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2013/4/26

17068

Subject: Interlace ppm file to RGB code

```
/*
  file : 3line-column.c
         which is modified from p2vec.c by GS Huang, which was modified from
ppm2rgb.c
  Program : ppm file format to ckt deint_tb.vhdl sim readable
       By: Chen, Blair Jian-Tin, Apr 25, 2000

*/

/*
  file : ppm2rgb.c
         which is modified from s4p3.c
  Program : ppm file format to ckt sim readable
       By: Chuang, Gene Chun-Hsiung, Jun 19

*/

#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

#define PPMP3 1    /* PPM picture format is P3 */
#define PPMP6 0    /* PPM picture format is P6 */


main()
{
  FILE *fpo,*fpi;
  int p3_p6n, inpicture, outputport;
  int k,y,j;
  int hi,wi, colormax;
  unsigned char rr, gg, bb;
  int irr, igg, ibb;
  char str[30];
  int inV_active, inH_active;
  unsigned char RGBarray[1920][3][3];
  unsigned char rrr, ggo, bbo;
  int xl, yl;


  setbuf(stdout,NULL);

  printf("Input picture is
0:comps(VGA),1:textvga,2:textSVGA,3:textXGA,4:VGAtestp\n");
  printf("or 5:XGAtest,6:GeneSVGA,7:GeneXGA,8:WStestp(1152*864),9:SXGAtestp\n");
  printf("or 10:SVGAtestp,11:Ingrid(720*576),12:FS_XGA,13:HS_XGA\n");
  printf("or
14:1280x960,15:1600,16:1920X1080,17:1920X1200,18:novashop,19:zebraSVGA\n");
  scanf("%d",&inpicture);

  switch (inpicture) {
        case 0:
                fpi = fopen ("comps.ppm", "r");
                inH_active = 640;
                inV_active = 480;
                break;
        case 1:
                fpi = fopen ("textvga.ppm", "r");
                inH_active = 640;
                inV_active = 480;
                break;
        case 2:
                fpi = fopen ("textSVGA.ppm", "r");
                inH_active = 800;
                inV_active = 600;
                break;
        case 3:
                fpi = fopen ("textXGA.ppm", "r");
                inH_active = 1024;
                inV_active = 768;
                break;
        case 4:
                fpi = fopen ("VGAtestp.ppm", "r");
                inH_active = 640;
                inV_active = 480;
               break;
        case 5:
                fpi = fopen ("XGAtest.ppm", "r");
                inH_active = 1024;
                inV_active = 768;
                break;
        case 6:
                fpi = fopen ("GeneSVGA.ppm", "r");
                inH_active = 800;
                inV_active = 600;
                break;
        case 7:
                fpi = fopen ("GeneXGA.ppm", "r");
                inH_active = 1024;
                inV_active = 768;
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/6/26

Subject: _____

File: D:\projects\Scaler\Oslo\deinterlace\Modify\C\3line-column.C 2000/4/26, 05:52:08PM

```
                break;
        case 8:
                fpi = fopen ("WStestp.ppm", "r");
                inH_active = 1152;
                inV_active = 864;
                break;
        case 9:
                fpi = fopen ("SXGAtestp.ppm", "r");
                inH_active = 1280;
                inV_active = 1024;
                break;
        case 10:
                fpi = fopen ("SVGAtestp.ppm", "r");
                inH_active = 800;
                inV_active = 600;
                break;
        case 11:
                fpi = fopen ("Ingrid.ppm", "r");
                inH_active = 720;
                inV_active = 576;
                break;
        case 12:
                fpi = fopen ("FS_XGA.ppm", "r");
                inH_active = 1024;
                inV_active = 768;
                break;
        case 13:
                fpi = fopen ("HS_XGA.ppm", "r");
                inH_active = 512;
                inV_active = 768;
                break;
        case 14:
                fpi = fopen ("1280X960.ppm", "r");
                inH_active = 1280;
                inV_active = 960;
                break;
        case 15:
                fpi = fopen ("1600.ppm", "r");
                inH_active = 1600;
                inV_active = 1280;
                break;
        case 16:
                fpi = fopen ("1920X1080.ppm", "r");
                inH_active = 1920;
                inV_active = 1080;
                break;
        case 17:
                fpi = fopen ("1920X1200.ppm", "r");
                inH_active = 1920;
                inV_active = 1200;
                break;
        case 18:
                fpi = fopen ("novashop.ppm", "r");
                inH_active = 720;
                inV_active = 480;
                break;
        case 19:
                fpi = fopen ("zebraSVGA.ppm", "r");
                inH_active = 800;
                inV_active = 600;
                break;
      default:
                fpi = fopen ("VGAtestp.ppm", "r");
                inH_active = 640;
                inV_active = 480;
                break;
}

printf("Output port no.: 1:one port, 2:two port, 3:three port\n");
scanf("%d",&outputport);


      if (fpi == (FILE *) NULL)
  {
    printf("Couldn't open the input file\n");
    exit(1);
  }
fpo = fopen ("testp.rgb", "w");
printf("The output file %s contents rr gg bb <cr>\n", "testp.rgb");

      if (fpo == (FILE *) NULL)
  {
    printf("Couldn't open %s for writing. \n", "testp.rgb");
    exit(0);
  }


/* Read ppm file's headers and write them to out file*/

fscanf(fpi,"%s",str);  printf("%s\n",str);
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2:/4/xt

Subject: _____

File: D:\projects\Scaler\Oslo\deinterlace\Modify\C\3line-column.C 2000/4/24, 05:52:08PM

```
  if (strcmp(str,"P6") == 0) {
        p3_p6n = PPMP6;
        printf("The input %s file will be converted to P3 ppm\n",str);
  } else { p3_p6n=PPMP3; }

  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s",str);
  fscanf(fpi,"%s",str);  printf("%s\n",str);

  if (p3_p6n == PPMP6) {
        fscanf(fpi,"%s",str);  printf("%s\n",str);
  }

  fscanf(fpi,"%d %d",&wi,&hi);
  printf("Original picture dimension: Width=%d, Height=%d \n",wi,hi);


  fscanf(fpi,"%d\n",&colormax);
  printf("color value max =%d \n",colormax);

  /* get RGB data for a pixel from the ppm file */

  for (y=0; y< hi; y++) {
     for (x=0; x< wi; x++) {

        if (p3_p6n == PPMP6) fscanf(fpi,"%c%c%c",&rr,&gg,&bb);
        else {
                fscanf(fpi,"%d%d%d",&irr,&igg,&ibb);
                rr = irr; gg = igg; bb = ibb;
             }

                if (y == 0) {

                RGBarray[x][0][0] = rr;
                RGBarray[x][0][1] = gg;
                RGBarray[x][0][2] = bb;

                } else if ((y%2) == 1) {

                RGBarray[x][1][0] = rr;
                RGBarray[x][1][1] = gg;
                RGBarray[x][1][2] = bb;

                } else {

                RGBarray[x][2][0] = rr;
                RGBarray[x][2][1] = gg;
                RGBarray[x][2][2] = bb;

                }

        }

        if ((y>0)&&((y%2) == 0))
        for (x1=0; x1< wi; x1++) {
                for (y1=0; y1<3; y1++) {

                        rro = RGBarray[x1][y1][0] ;
                        ggo = RGBarray[x1][y1][1] ;
                        bbo = RGBarray[x1][y1][2] ;

                        if ( outputport == 1)
                                fprintf(fpo,"%02x %02x %02x\n",rro,ggo,bbo);

                        else if ( outputport == 2)
                        {
                                j=x1%2;
                                if(j==1) fprintf(fpo,"%02x %02x                 !
%02x\n",rro,ggo,bbo);
                                else fprintf(fpo,"%02x %02x %02x ",rro,ggo,bbo);
                        }

                        else {
                                if (y1==2) fprintf(fpo,"%02x %02x
%02x\n",rro,ggo,bbo);
                                else fprintf(fpo,"%02x %02x %02x ",rro,ggo,bbo);
                                }


                }

        RGBarray[x1][0][0] = RGBarray[x1][2][0];
        RGBarray[x1][0][1] = RGBarray[x1][2][1];
        RGBarray[x1][0][2] = RGBarray[x1][2][2];
        }
```

isclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2003/4/26

Subject: _____

File: D:\projects\Scaler\Oslo\deinterlace\Modify\C\3line-column.C 2000/4/26, 05:52:08PM

```
    )

    printf("The output file is %s which contents rearranged rr gg bb <cr>\n",
"testp.rgb");

    fclose(fpi);
    fclose(fpo);

} /* END of main */
```

Disclosed To And Understood By Me

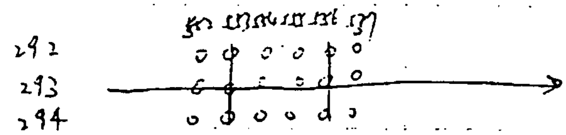
Name	Date	Signature: blair
Name	Date	Date: 2000/4/26

17072

Subject: Debug for de-interlace

頭髮白點 (553, 293) (556, 293)

→ 293 × 720 + 553 (556)+1 = 211514th & 211517th



⇒ 292 × 720 + 553 ~ 558 = 210793 ~ 210798

293 × 720 + 553 ~ 558 = 211513 ~ 211518

294 × 720 + 553 ~ 558 = 212233 ~ 212238

出 input 用 UltraEdit / Search / Word Count

553 ⟶ 558

line 294th	255	246	208	255	247	209
293th	255	168	63	255	165	39
292th	252	[illegible]	255	255	255	255

output

line 294th	255	246	208	255	247	209
293th	255	130	[illegible]	255	147	31
292th	255	[illegible]	255	255	255	255

點數前面 ×3，直接 RGB 皆納入

直接用 RGB 查較快

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2000/4/27

Subject:

:93 是 第 293 條 ∴ line = 292/2 ×720 + 553~558

= 105673 ~ 105678 前面

293 " 294 " ∴ line = 105673 ~ 105678 中間

294 " 295 " ∴ line = 294/2 ×720 + 553 ~ 558

= 106393 ~ 106398 或 292.293 兩條 (input)

input		552	553	554	555	556	557
line	292	24 19 39	02 21 24	09 1f 1f	12 28 28	1c 24 24	20 29 29
	293	e3 ff ff	ec ff ff	f3 ff ff	f5 ff ff	e9 ff ff	b2 d6 dd
	294	23 36 36	cd 1f 25	08 1b 21	13 25 2b	1c 23 26	20 28 2b

Output		552	553	554	555	556	557	
line	292	24 19 39	02 21 24	09 1f 1f	12 28 28	1c 24 24	20 29 29	→ 同 input
	293	23 37 39	ec ff ff	10 24 27	12 26 29	e9 ff ff	24 2c 2e	
	294	23 36 36	cd 1f 25	08 1b 21	13 25 2b	1c 23 26	20 28 2b	→ 同 input.

同 input 同 input

2e dc db = 295

point (553, 293) : diff = total 52E (df e0 da = 297); e3 e0 e0 = 2A3, total 4F9, c9 c9 c4 = 2b6; c8 c6 c8 = 2b3, total 4F9, e9 e4 de = 2A6

=74F9 = 4F9 . 兩數相同 未 detect 到

nova 在 THDuration = 20 時, 而 140 即開始出現條紋

zebra 則變模糊

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 200/4/28

17071

Subject: RGB to PPM c code modify

File: D:\projects\Scaler\Oslo\deinterlace\Modify\C\rgb2ppmg.c 2000/4/28, 05:33:42PM

```
/*-----------------------------------------
*  translate scale.v output R,G,B 24 bits  *
*  (output.txt) to 3 characters           *
-----------------------------------------*/

#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

main()
{
  FILE *fpo,*fpi;
  unsigned char c[12];
  int x,y;
  int sdhi=600;
  int sdwi=800;
  int tmp[2],tmp1;
  int i,j;
  int inpicture;

  char str[30];
  int RGBarray[5760];
  int k,l,m,xl;


  fpi = fopen ("output.rgb", "r" );
  if (fpi == (FILE *) NULL)
  {
    printf("Couldn't open the input file\n");
    exit(1);
  }

  fpo = fopen ("output.ppm", "w");
  if (fpo == (FILE *) NULL)
  {
    printf("Couldn't open %s for writing. \n", "output.ppm");
    exit(0);
  }

  printf("The file output.txt is --
0:VGA,1:SVGA,2:XGA,3:1152x864,4:SXGA,5:720x576,6:720x480,7:special\n");
  scanf("%d",&inpicture);

  switch (inpicture) {
        case 0:
                sdwi=640;
                sdhi=480;
                break;
        case 1:
                sdwi=800;
                sdhi=600;
                break;
        case 2:
                sdwi=1024;
                sdhi=768;
                break;
        case 3:
                sdwi=1152;
                sdhi=864;
                break;
        case 4:
                sdwi=1280;
                sdhi=1024;
                break;
        case 5:
                sdwi=720;
                sdhi=576;
                break;
        case 6:
                sdwi=720;
                sdhi=480;
                break;
        case 7:
                printf("Width is \n",);
                scanf("%d",&sdwi);
                printf("Height is \n",);
                scanf("%d",&sdhi);
                break;
        default:
                sdwi=1024;
                sdhi=768;
                 break;
  }


  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
```

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2000/4/28

Subject: _____

File: D:\projects\Scaler\Oslo\deinterlace\Modify\C\rgb2ppmg.c 2000/4/28, 05:33:42PM

```
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);
  fscanf(fpi,"%s\n",str);


  /*  write ppm file's headers to output.ppm */
  fprintf(fpo,"P3\n");
  fprintf(fpo,"# By: Gene Chuang  trumpion us+ (copy right)\n");
  fprintf(fpo,"%d %d\n",sdwi,sdhi);
  fprintf(fpo,"255");

  fprintf(fpo,"\n");

  /*  read in output.rgb and translate it into char  */
  for(y=0;y<sdhi;y++)
  {
        i=0;


    for(x=0;x<sdwi;x++)
    {
        m=x%6;


fscanf(fpi,"%c%c%c%c%c%c%c%c%c%c%c%c\n",&r[0],&r[1],&r[2],&r[3],&r[4],&r[5],&r[6
],&r[7],&r[8],&r[9],&r[10],&r[11]);
      for(i=0;i<6;i++)
      {
       tmp[0]=r[i*2];
           tmp[1]=r[i*2+1];


           for (j=0;j<2;j++) {
       if(tmp[j]>=48 && tmp[j]<=57)  tmp[j]=tmp[j]-48;    /* char 0 ~9 */
       else if(tmp[j]>=97 && tmp[j]<=102)  tmp[j]=tmp[j]-87;   /* char a ~ f */
           else if(tmp[j]>=65 && tmp[j]<=70)  tmp[j]=tmp[j]-55;  /* char A ~ F
*/
       else  tmp[j]=0;
                }

/*          fprintf(fpo,"r is %c%c; tmp is %d %d ", r[i*2], r[i*2+1], tmp[0],
tmp[1]);
                    */

           tmp1=16*tmp[0] + tmp[1];

        if(i<3)
        {
                        fprintf(fpo,"%03d ",tmp1);

                        if((m==5)&&(i==2))
                        fprintf(fpo,"\n");

        } else {

                                RGBarray[l]= tmp1;
                                l++;

                                if ((x==(sdwi-1))&&(i==5))
                        {
                                for (xl=0;xl<l;xl++)
                                {
                                fprintf(fpo,"%3d ",RGBarray[xl]);

                                k=xl%18;
                                        if(k==17)
                                        fprintf(fpo,"\n");
                                        }

                                }
                        }
      }
     }

  }
  fclose(fpi);
  fclose(fpo);


  printf("The output file is %s which contents de-interlaced ppm image<cr>\n",
"output.ppm");

}
```

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2000/4/28

17070

Subject: deinterlace test bench

```
IBRARY ieee ;
IBRARY oslo_lib ;

SE std.textio.ALL ;
SE ieee.std_logic_1164.ALL ;
SE ieee.std_logic_arith.ALL ;

SE oslo_lib.oslo_components_pkg.ALL ;
SE oslo_lib.std_rw.ALL;

- escalade ports -begin
ntity deint_tb is
d deint_tb;

- escalade ports -end

CHITECTURE behav_tb OF deint_tb IS

   CONSTANT clockperiod : time := 16 ns;
   CONSTANT tpd         : time := 1 ns;


ignal   clk         : std_logic := '0';

ignal   rstn        : std_logic;
ignal   powerdown   : std_logic;
gnal    pv_active   : std_logic;
gnal    swap_hs     : std_logic;
gnal    abuf_data   : std_logic_vector(23 downto 0);
ignal   bbuf_data   : std_logic_vector(23 downto 0);
ignal   cbuf_data   : std_logic_vector(23 downto 0);
   l    THDmotion   : std_logic_vector(7 downto 0);

ignal   LA_out      : std_logic_vector(23 downto 0);
gnal    LB_out      : std_logic_vector(23 downto 0);
ignal   LC_out      : std_logic_vector(23 downto 0);
gnal    finished    : std_logic := '0';

                -- main output SIGNALs here

gnal Aout_val          : std_logic_vector(23 downto 0);
gnal Bout_val          : std_logic_vector(23 downto 0);
gnal Cout_val          : std_logic_vector(23 downto 0);
```

ge: 1

isclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2003/4/28

Subject: _____ 17077

D:\projects\Scaler\Oslo\deinterlace\Modify\simtest\proj_test\deint_tb\behav_t

```
BEGIN

dut: deint
PORT MAP(
     rstn,
     powerdown,
     clk,
     pv_active,
     swap_hs,
     abuf_data,
     bbuf_data,
     cbuf_data,
     THDmotion,
     LA_out,
     LB_out,
     LC_out
     );


    PROCESS
               -- textio input file and output file

        USE std.textio.ALL;
        constant INFILE: string := "d:\projects\ppm\testp.rgb";
        FILE     finp      : text IS IN INFILE;

                 -- main input SIGNALs here

        VARIABLE Rin_vala       : std_logic_vector(7 downto 0);
        VARIABLE Gin_vala       : std_logic_vector(7 downto 0);
        VARIABLE Bin_vala       : std_logic_vector(7 downto 0);
        VARIABLE Rin_valb       : std_logic_vector(7 downto 0);
        VARIABLE Gin_valb       : std_logic_vector(7 downto 0);
        VARIABLE Bin_valb       : std_logic_vector(7 downto 0);
        VARIABLE Rin_valc       : std_logic_vector(7 downto 0);
        VARIABLE Gin_valc       : std_logic_vector(7 downto 0);
        VARIABLE Bin_valc       : std_logic_vector(7 downto 0);

        VARIABLE li, lo         : line;
        VARIABLE comment        : boolean := false;
        VARIABLE ct             : time;


        constant OUTFILE: string := "d:\projects\ppm\output.rgb";
        FILE     fout      : text IS out OUTFILE;
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2003/4/28

17078
ubject: _____

```
BEGIN
    WAIT FOR 0 ns;
    WHILE NOT endfile(finp) LOOP
        mreadline(finp, li, comment);
        IF NOT comment THEN
            mread(li, Rin_vala, h);
            mread(li, Gin_vala, h);
            mread(li, Bin_vala, h);
            mread(li, Rin_valb, h);
            mread(li, Gin_valb, h);
            mread(li, Bin_valb, h);
            mread(li, Rin_valc, h);


             mread(li, Gin_valc, h);
             mread(li, Bin_valc, h);


        -- main input SIGNALs here

        abuf_data <= Rin_vala & Gin_vala & Bin_vala;
        bbuf_data <= Rin_valb & Gin_valb & Bin_valb;
        cbuf_data <= Rin_valc & Gin_valc & Bin_valc;


        END IF;

        Aout_val <= LA_out;
        Bout_val <= LB_out;
        Cout_val <= LC_out;

        mwrite(lo, Aout_val, right, 6, h);
        mwrite(lo, Bout_val, right, 6, h);
        writeline(fout, lo);

    wait for 1 * CLOCKPERIOD;

    END LOOP;
    WAIT;
  END PROCESS ;

ocess
egin
  clk <= '0';
  loop
      wait for (CLOCKPERIOD * 0.5);
      clk <= '1';
      wait for (CLOCKPERIOD * 0.5);
      clk <= '0';
  end loop;
  wait;
nd process;
```

closed To And Understood By Me

Name	Date
Name	Date

Signature: _____

Date: 2003/4/28

Subject: _____

```
    powerdown <= '0';
    swap_hs <= '0';

    THDmotion <= "00100000";

    rstn <= '1';
    pv_active <= '1';

:D behav_tb;
```

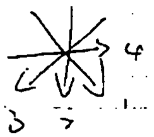

THDmotion 在 20(H) 時，動態之 waveshop
也可得到滿意的結果
但靜態之 zebra 則呈現較模糊的情形
(如 21: 說明之)
但若以 11 則畫面縫隙很大
水波紋較少，特技較模糊
但被馬則看不太出來

Disclosed To And Understood By Me

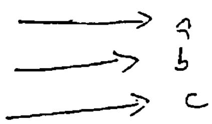
Name	Date	Signature: blair
Name	Date	Date: 2001/4/28

17080

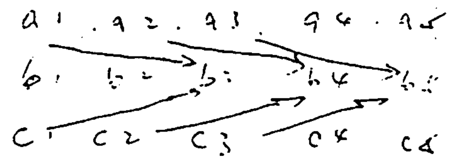
Subject: De-interlace Algorithm



if 4 方向 compare，之如何？



if a.c 之間找 threshold > 6



60 frame/sec => 16.66 ms/frame

以 1920×1200/frame 來算

=> 14.4676 ns/dot

17.36 μs/line

• a.b.c 取樣 n bit 數

$a_1 (a_2\ a_3\ a_4\ a_5\ a_6\ a_7\ a_8\ a_9)\ a_{10}$

$b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ b_7\ b_8\ b_9\ b_{10}$

$c_1 (c_2\ c_3\ c_4\ c_5\ c_6\ c_7\ c_8\ c_9)\ c_{10}$

if $(a_1 + a_2 + \cdots + a_9) - (c_2 + c_3 + \cdots + c_{10})$
$(a_1 + \cdots + a_9) - (c_1 + \cdots + c_9)$
$(a_2 + \cdots + a_{10}) - (c_1 + \cdots + c_9)$

minimum => [illegible]

Disclosed To And Understood By Me

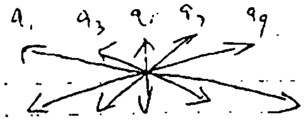
Name Date

Name Date

Signature: [illegible]

Date: 2002/8/2

Subject:



先 check $(a_n - c_n) \geq (a_n - b_n)$ or $(a_n - c_m) \geq (b_{\frac{n+m}{2}} - c_m)$

if $\geq$ => b_n or $b_{\frac{n+m}{2}}$ 不變

or 3 選 2 or 5 選 中 $\geq$ => 不變

若 ~~中~~ $<$ => ~~can~~ move and cannot judge

=> 原法 ~~米~~ 選 minimum

Algorithm => if ~~b_n — b_{n+j}~~
$(a_n - b_n)$ … $(a_{n+j} - b_{n+j})$ ~~<~~ $(a_n - c_n)$ … $(a_{n+j} - c_{n+j})$

then ~~move and~~ ~~a~~ $b_n \Leftarrow b_m$ … $b_{n+j} \Leftarrow b_{m+j}$

else 原方法 (old algorithm)

The simplest way is to check ~~1 2 3~~ a_1 a_2 a_3 a_4 a_5
~~4 5 6~~ b_1 b_2 b_3 b_4 b_5
~~7 8 9~~ c_1 c_2 c_3 c_4 c_5

if ~~for~~ $n = 1 \sim 5$ ~~[illegible]~~ $a_n - b_n < a_n - c_n$ 有 3 個以上成立

then still picture (but for ~~移~~ moving picture, it will show still for some case ∴ 5 → 4 試看)

~~else~~

or 4 取 2 →

else b_2 = old algorithm or 3 取 1 true => still

or 5 取 2 (or 1)

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2002/05/3

17082

Subject: Jpeg pump result

File: D:\Jpegtest1.txt 2000/5/3, 02:57:11PM

```
( Start Of Image )
( APP0 Marker
  Length: 16
  Version: 1.1
  Density Unit:   (aspect ratio)
  X Density: 1
  Y Density: 1
  Thumbnail Width: 0
  Thumbnail Height: 0
)
( Define Quantization Table
  Length:  67
  Table Index: 0
  Table Precision: 0
  Table Values:
        11 8 8 10 8 7 11 10
        9 10 13 12 11 13 17 28
        18 17 15 15 17 34 25 26
        20 28 41 36 43 42 40 36
        39 39 45 50 64 55 45 48
        61 48 39 39 56 76 57 61
        67 69 72 73 72 43 54 79
        85 78 70 84 64 71 72 69

)
( Define Quantization Table
  Length:  67
  Table Index: 1
  Table Precision: 0
  Table Values:
        12 13 13 17 15 17 33 15
        18 33 69 44 39 46 69 69
        69 69 69 69 69 69 69 69
        69 69 69 69 69 69 69 69
        69 69 69 69 69 69 69 69
        69 69 69 69 69 69 69 69
        69 69 69 69 69 69 69 69
        69 69 69 69 69 69 69 69

)
( Start Of Frame
  Type: Baseline Huffman
  Length: 17
  Precision: 8
  Height: 2362
  Width: 3478
  Component Count: 3
   Component 1
    Horizontal Frequency: 2
    Vertical Frequency: 2
    Quantization Table: 0
   Component 2
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
   Component 3
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
)
( Define Huffman Table
  Length: 31
   Table Index 0
    Table Class: DC
    Code Counts: 0 1 5 1 1 1 1 1 1 0 0 0 0 0 0 0
    Code Values: 0 1 2 3 4 5 6 7 8 9 a b
)
( Define Huffman Table
  Length: 181
   Table Index 0
    Table Class: AC
    Code Counts: 0 2 1 3 3 2 4 3 5 5 4 4 0 0 1 7d
    Code Values:  1 2 3 0 4 11 5 12 21 31 41 6 13 51 61 7
                  22 71 14 32 81 91 a1 8 23 42 b1 c1 15 52 d1 f0
                  24 33 62 72 82 9 a 16 17 18 19 1a 25 26 27 28
                  29 2a 34 35 36 37 38 39 3a 43 44 45 46 47 48 49
                  4a 53 54 55 56 57 58 59 5a 63 64 65 66 67 68 69
                  6a 73 74 75 76 77 78 79 7a 83 84 85 86 87 88 89
                  8a 92 93 94 95 96 97 98 99 a2 a3 a4 a5 a6 a7
                  a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3 c4 c5
                  c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da e1 e2
                  e3 e4 e5 e6 e7 e8 e9 ea f1 f2 f3 f4 f5 f6 f7 f8
                  f9 fa
)
( Define Huffman Table
  Length: 31
   Table Index 1
    Table Class: DC
    Code Counts: 0 3 1 1 1 1 1 1 1 1 0 0 0 0 0 0
    Code Values:  0 1 2 3 4 5 6 7 8 9 a b
)
( Define Huffman Table
```

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2000/5/3

17083

Subject: _____

File: D:\Jpegtest\1.txt 2000/5/3, 02:57:11PM

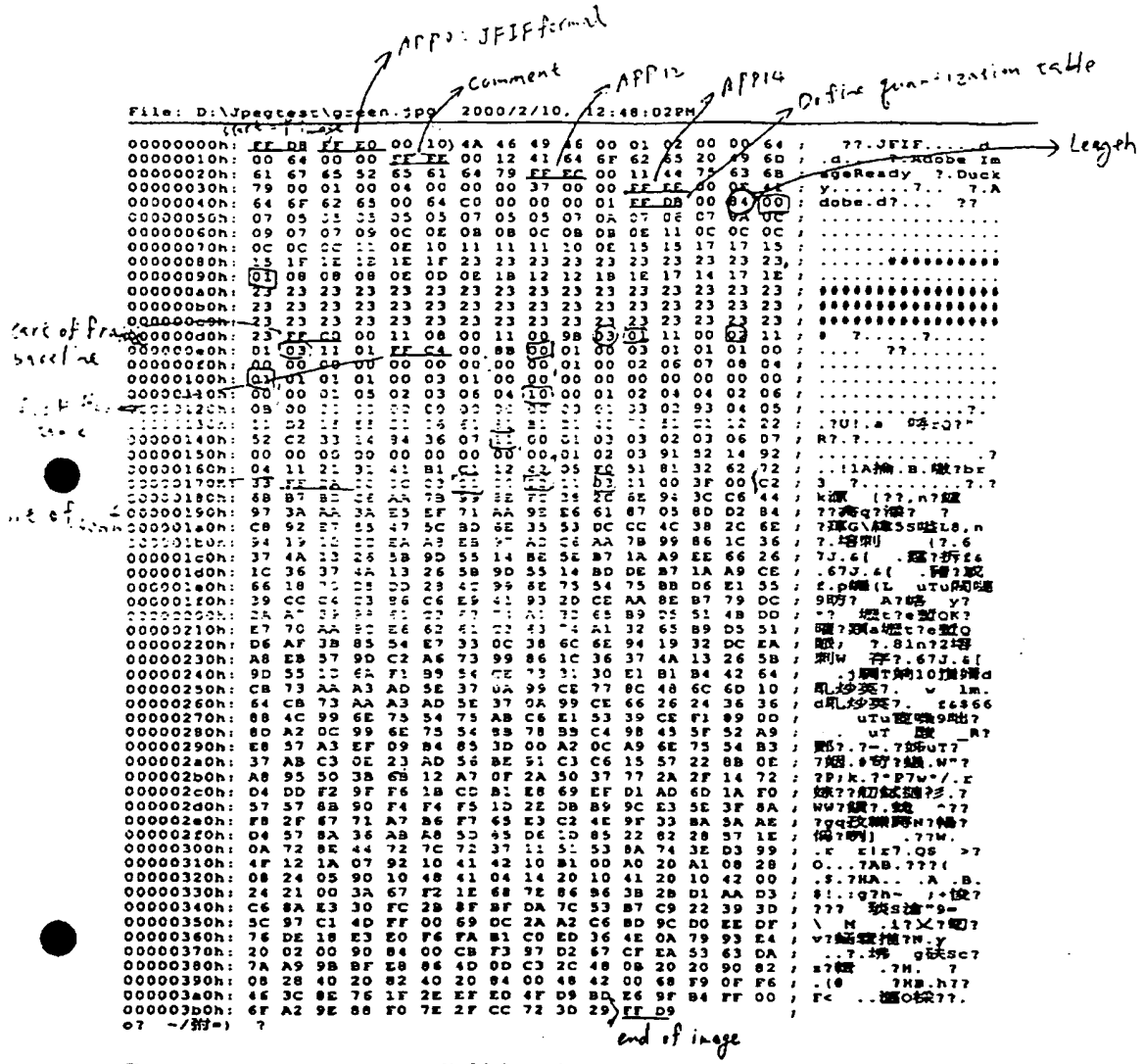
```
  Length: 181
   Table Index 1
    Table Class: AC
    Code Counts: 0 2 1 2 4 4 3 4 7 5 4 4 0 1 2 77
    Code Values:  0  1  2  3 11  4  5 21 31  6 12 41 51  7 61 71
                 13 22 32 81  8 14 42 91 a1 b1 c1  9 23 33 52 f0
                 15 62 72 d1  a 16 24 34 e1 25 f1 17 18 19 1a 26
                 27 28 29 2a 35 36 37 38 39 3a 43 44 45 46 47 48
                 49 4a 53 54 55 56 57 58 59 5a 63 64 65 66 67 68
                 69 6a 73 74 75 76 77 78 79 7a 82 83 84 85 86 87
                 88 89 8a 92 93 94 95 96 97 98 99 9a a2 a3 a4 a5
                 a6 a7 a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3
                 c4 c5 c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da
                 e2 e3 e4 e5 e6 e7 e8 e9 ea f2 f3 f4 f5 f6 f7 f8
                 f9 fa
)
( Start Of Scan
  Length:  12
  Scan Count: 3
   Component ID: 1
    AC Entropy Table: 0
    DC Entropy Table: 0
   Component ID: 2
    AC Entropy Table: 1
    DC Entropy Table: 1
   Component ID: 3
    AC Entropy Table: 1
    DC Entropy Table: 1
  Spectral Selection Start: 0
  Spectral Selection End: 63
  Sucessive Approximation High: 0
  Sucessive Approximation Low: 0
)

( End Of Image )
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/5/3

17084

Subject: Jpeg wide scanning



Disclosed To And Understood By Me

Name Date Signature:

Name Date Date: 2000/5/8

Subject: _____

```
( APPE Marker
  Length: 14
Adobe
)
( Define Quantization Table
  Length:  132
  Table Index: 0
  Table Precision: 0
  Table Values:
        7 5 5 5 5 5 7 5
        5 7 10 7 6 7 10 12
        9 7 7 9 12 14 11 11
        12 11 11 14 17 12 12 12
        12 12 12 17 14 16 17 17
        17 16 14 21 21 23 23 21
        21 31 30 30 30 31 35 35
        35 35 35 35 35 35 35 35
  Table Index: 1
  Table Precision: 0
  Table Values:
        8 8 8 14 13 14 27 18
        18 27 30 23 20 23 30 35
        35 35 35 35 35 35 35 35
        35 35 35 35 35 35 35 35
        35 35 35 35 35 35 35 35
        35 35 35 35 35 35 35 35
        35 35 35 35 35 35 35 35
        35 35 35 35 35 35 35 35
)
( Start Of Frame
  Type: Baseline (Huffman)
  Length: 17
  Precision: 8
  Height:
  Width:
  Component Count: 3
   Component 1
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 0
   Component 2
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
   Component 3
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
)
( Define Huffman Table
  Length: 139
   Table Index 0
    Table Class: DC
    Code Counts: 1 0 3 1 1 1 0 0 0 0 0 0 0 0 0 0
    Code Values: 0 1 2 6 7 8 4 5
   Table Index 1
    Table Class: DC
    Code Counts: 1 1 1 0 3 1 0 0 0 0 0 0 0 0 0 0
    Code Values: 0 1 5 2 3 6 4
   Table Index 0
    Table Class: AC
    Code Counts: 0 1 2 4 4 2 6 b 0 0 0 0 0 0 0 0
    Code Values: 0 1 3 2 93 4 5 11 d2 15 55 21 6 61 81 b1
                 d1 42 72 51 c1 12 22 52 c2 33 14 84 36 7
   Table Index 1
    Table Class: AC
    Code Counts: 0 1 3 3 2 3 6 7 0 0 0 0 0 0 0 0
    Code Values: 0 1 2 3 91 52 14 92 a 23 21 31 41 b1 c1 12
                 42 5 f0 51 81 32 62 72 33
)
( Start Of Scan
  Length: 12
  Scan Count: 3
```



```
   Component ID: 1
    AC Entropy Table: 0
    DC Entropy Table: 0
   Component ID: 2
    AC Entropy Table: 1
    DC Entropy Table: 1
   Component ID: 3
    AC Entropy Table: 1
    DC Entropy Table: 1
  Spectral Selection Start: 0
  Spectral Selection End: 63
  Sucessive Approximation High: 0
  Sucessive Approximation Low: 0
)

( End Of Image )
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2001/5/9

17086

Object: _____

File: D:\Jpegtest\blutxtr2.jpg 1998/12/29, 03:18:56PM

```
00000000h: FF D8 FF E0 00 10 4A 46 49 46 00 01 01 00 00 01 ;   ??.JFIF.....
00000010h: 00 01 00 00 FF DB 00 43 00 0B 07 08 09 08 07 0B ; ....  ?C........
00000020h: 09 09 09 0C 0B 0B 0D 10 1A 11 10 0F 0F 10 20 17 ; .............. .
00000030h: 18 13 1A 26 22 28 28 28 26 22 25 24 2A 30 3D 33 2A ; ...&"((&"%$*0=3*
00000040h: 2D 39 2E 24 25 35 48 35 39 3F 41 44 45 44 29 33 ; -9.$%5H59?ADED)3
00000050h: 4B 50 4A 42 4F 3D 43 44 41 FF DB 00 43 01 0B 0C ; KPJBO=CDA  ?C...
00000060h: 0C 10 0E 10 1F 11 11 1F 41 2C 25 2C 41 41 41 41 ; ........A,%,AAAA
00000070h: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 ; AAAAAAAAAAAAAAAA
00000080h: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 ; AAAAAAAAAAAAAAAA
00000090h: 41 41 41 41 41 41 41 41 41 41 41 41 41 41 FF C0 ; AAAAAAAAAAAAAA  ?
000000a0h: 00 11 08 00 40 00 40 03 01 22 00 02 11 01 03 11 ; ....@.@..".....
000000b0h: 01 FF C4 00 1F 00 00 01 05 01 01 01 01 01 01 00 ; .  ?............
000000c0h: 00 00 00 00 00 00 00 01 02 03 04 05 06 07 08 09 ; ................
000000d0h: 0A 0B FF C4 00 B5 10 00 02 01 03 03 02 04 03 05 ; ..  ??..........
000000e0h: 05 04 04 00 00 01 7D 01 02 03 00 04 11 05 12 21 ; ......}........!
000000f0h: 31 41 06 13 51 61 07 22 71 14 32 81 91 A1 08 23 ; 1A..Qa."q.2??#
00000100h: 42 B1 C1 15 52 D1 F0 24 33 62 72 82 09 0A 16 17 ; B?R?$3br?....
00000110h: 18 19 1A 25 26 27 28 29 2A 34 35 36 37 38 39 3A ; ...%&'()*456789:
00000120h: 43 44 45 46 47 48 49 4A 53 54 55 56 57 58 59 5A ; CDEFGHIJSTUVWXYZ
00000130h: 63 64 65 66 67 68 69 6A 73 74 75 76 77 78 79 7A ; cdefghijstuvwxyz
00000140h: 83 84 85 86 87 88 89 8A 92 93 94 95 96 97 98 99 ; ????????
00000150h: 9A A2 A3 A4 A5 A6 A7 A8 A9 AA B2 B3 B4 B5 B6 B7 ;   ?????????
00000160h: B8 B9 BA C2 C3 C4 C5 C6 C7 C8 C9 CA D2 D3 D4 D5 ; ????????????
00000170h: D6 D7 D8 D9 DA E1 E2 E3 E4 E5 E6 E7 E8 E9 EA F1 ; ????????????
00000180h: F2 F3 F4 F5 F6 F7 F8 F9 FA FF C4 00 1F 01 00 03 ; ??????  ?.....
00000190h: 01 01 01 01 01 01 01 01 01 00 00 00 00 00 00 01 ; ................
000001a0h: 02 03 04 05 06 07 08 09 0A 0B FF C4 00 B5 11 00 ; ..........  ??.
000001b0h: 02 01 02 04 04 03 04 07 05 04 04 00 01 02 77 00 ; ..............w.
000001c0h: 01 02 03 11 04 05 21 31 06 12 41 51 07 61 71 13 ; ......!1..AQ.aq.
000001d0h: 22 32 81 08 14 42 91 A1 B1 C1 09 23 33 52 F0 15 ; "2?.B  ???#3R?
000001e0h: 62 72 D1 0A 16 24 34 E1 25 F1 17 18 19 1A 26 27 ; br?.$4??...&'
000001f0h: 28 29 2A 35 36 37 38 39 3A 43 44 45 46 47 48 49 ; ()*56789:CDEFGHI
00000200h: 4A 53 54 55 56 57 58 59 5A 63 64 65 66 67 68 69 ; JSTUVWXYZcdefghi
00000210h: 6A 73 74 75 76 77 78 79 7A 82 83 84 85 86 87 88 ; jstuvwxyz????
00000220h: 89 8A 92 93 94 95 96 97 98 99 9A A2 A3 A4 A5 A6 ; ?????  ??????
00000230h: A7 A8 A9 AA B2 B3 B4 B5 B6 B7 B8 B9 BA C2 C3 C4 ; ????????????
00000240h: C5 C6 C7 C8 C9 CA D2 D3 D4 D5 D6 D7 D8 D9 DA E2 ; ????????????
00000250h: E3 E4 E5 E6 E7 E8 E9 EA F2 F3 F4 F5 F6 F7 F8 F9 ; ????????????
00000260h: FA FF DA 00 0C 03 01 00 02 11 03 11 00 3F 00 E8 ; ??.........?.?
00000270h: C8 C7 60 69 41 E9 8E 29 81 9E 79 14 A1 F9 C1 15 ;  `iA?)?y.?/?
00000280h: E8 92 39 83 67 A9 A4 21 80 18 6A 37 72 29 49 A0 ; ?9  ?g? .)?!j7r)I?
00000290h: 66 10 [illegible] E4 D2 9D C3 [illegible] 35 [illegible] 73 FF 00 D6 A6 92 ; f..??E??5+s  .??E?
000002a0h: 78 A0 [illegible] E7 3E F4 98 1C 70 [illegible] 26 F3 8A 53 CF 6A ; x???.p.&?S?J
000002b0h: 00 46 5C 1C E6 90 8F 94 11 52 75 C7 6A 00 F9 88 ; .F\.??.Ru .?
000002c0h: CE 45 C0 45 BB 20 D2 E3 A7 34 E9 14 03 4D 39 C0 ; ?E?E?? .M9?
000002d0h: C7 34 [illegible] 63 3C D2 1F A2 29 [illegible] 1F 4A 72 C6 47 34 ; ?.c<???z.Jq?G4
000002e0h: 00 D5 C1 A3 1B FE 23 4E 2B 81 D7 1C D0 7A D0 00 ; .??N+ .??z?
000002f0h: 38 A6 86 C3 FA 8F 6A 77 EB 49 B4 1E 7A 66 80 06 ; 8?? wI?zf .
00000300h: 20 9E 2B C6 47 E3 41 5F 5E DD 29 39 1E D4 00 72 ;  ?+?G?A_^?9.?r
00000310h: 09 A6 49 C1 02 A4 CE 5B 3D 68 64 0C 41 A0 06 96 ; .?I?.??[=hd.A??
00000320h: 05 47 6A 32 0E 28 0B C6 3B A4 23 9E 4D 00 3F AF ; .Gj2.(.??  .??
00000330h: F0 D2 8E 07 14 CC FA 1C D2 80 DF 85 00 38 D2 13 ; ???.??.??.8?
00000340h: 8E D4 84 90 47 71 43 83 D6 80 14 B0 3D A8 C9 07 ;   ?GqC  .?=?.
00000350h: D6 99 CE E9 39 FA 50 04 BD 4E 71 4D 65 E3 A7 4A ; ?? ?9 .?P.?NqMe??J
00000360h: 41 BB 14 07 C7 07 AD 00 7F FF D9                ; A?.??  ?
```

File: D:\Jpegtest\blutxtr2dump.txt 2000/5/8, 10:24:00AM

```
( Start Of Image )
( APP0 Marker
  Length: 16
  Version: 1.1
  Density Unit:  (aspect ratio)
  X Density: 1
  Y Density: 1
  Thumbnail Width: 0
  Thumbnail Height: 0
)
( Define Quantization Table
  Length:  67
  Table Index: 0
  Table Precision: 0
  Table Values:
        11 7 8 9 8 7 11 9
        9 9 12 11 11 12 16 26
        17 16 15 15 16 32 24 24
        [illegible]
        37 36 42 48 61 51 42 45
        57 46 36 37 53 72 53 57
        63 65 68 69 68 41 51 73
        80 74 66 79 61 67 68 65

)
( Define Quantization Table
  Length:  67
  Table Index: 1
  Table Precision: 0
  Table Values:
        11 12 12 16 14 16 31 17
        17 31 65 44 37 44 65 65
        65 65 65 65 65 65 65 65
        65 65 65 65 65 65 65 65
```

closed To And Understood By Me

Name _____ Date _____ Signature: _____

Name _____ Date _____ Date: _____

Subject: _____ 17087

```
        65 65 65 65 65 65 65 65
        65 65 65 65 65 65 65 65
        65 65 65 65 65 65 65 65
        65 65 65 65 65 65 65 65
)
( Start Of Frame
  Type: Baseline (Huffman)
  Length: 17
  Precision: 8
  Height: 64
  Width: 64
  Component Count: 3
   Component 1
    Horizontal Frequency: 2
    Vertical Frequency: 2
    Quantization Table: 0
   Component 2
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
   Component 3
    Horizontal Frequency: 1
    Vertical Frequency: 1
    Quantization Table: 1
)
( Define Huffman Table
  Length: 31
   Table Index 0
    Table Class: DC
    Code Counts: 0 1 5 1 1 1 1 1 0 0 0 0 0 0 0
    Code Values:  0 1 2 3 4 5 6 7 8 9 a b
)
( Define Huffman Table
  Length: 181
   Table Index 0
    Table Class: AC
    Code Counts: 0 2 1 3 3 2 4 3 5 5 4 4 0 0 1 7d
    Code Values:  1 2 3 0 4 11 5 12 21 31 41 6 13 51 61 7
                 22 71 14 32 81 91 a1 8 23 42 b1 c1 15 52 d1 f0
                 24 33 62 72 82 9 a 16 17 18 19 1a 25 26 27 28
                 29 2a 34 35 36 37 38 39 3a 43 44 45 46 47 48 49
                 4a 53 54 55 56 57 58 59 5a 63 64 65 66 67 68 69
                 6a 73 74 75 76 77 78 79 7a 83 84 85 86 87 88 89
                 8a 92 93 94 95 96 97 98 99 9a a2 a3 a4 a5 a6 a7
                 a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3 c4 c5
                 c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da e1 e2
                 e3 e4 e5 e6 e7 e8 e9 ea f1 f2 f3 f4 f5 f6 f7 f8
                 f9 fa
)
( Define Huffman Table
  Length: 31
   Table Index 1
    Table Class: DC
    Code Counts: 0 3 1 1 1 1 1 1 1 1 0 0 0 0 0
    Code Values:  0 1 2 3 4 5 6 7 8 9 a b
)
( Define Huffman Table
  Length: 181
   Table Index 1
    Table Class: AC
    Code Counts: 0 2 1 2 4 4 3 4 7 5 4 4 0 1 2 77
    Code Values:  0 1 2 3 11 4 5 21 31 6 12 41 51 7 61 71
                 13 22 32 81 8 14 42 91 a1 b1 c1 9 23 33 52 f0
                 15 62 72 d1 a 16 24 34 e1 25 f1 17 18 19 1a 26
                 27 28 29 2a 35 36 37 38 39 3a 43 44 45 46 47 48
                 49 4a 53 54 55 56 57 58 59 5a 63 64 65 66 67 68
                 69 6a 73 74 75 76 77 78 79 7a 82 83 84 85 86 87
                 88 89 8a 92 93 94 95 96 97 98 99 9a a2 a3 a4 a5
                 a6 a7 a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3
                 c4 c5 c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da
                 e2 e3 e4 e5 e6 e7 e8 e9 ea f2 f3 f4 f5 f6 f7 f8
                 f9 fa
)
( Start Of Scan
  Length:  12
  Scan Count: 3
   Component ID: 1
    AC Entropy Table: 0
    DC Entropy Table: 0
   Component ID: 2
    AC Entropy Table: 1
    DC Entropy Table: 1
   Component ID: 3
    AC Entropy Table: 1
    DC Entropy Table: 1
  Spectral Selection Start: 0
  Spectral Selection End: 63
  Sucessive Approximation High: 0
  Sucessive Approximation Low: 0
)
( End Of Image )
```

sclosed To And Understood By Me

Name _____ Date _____ Signature: blair

Name _____ Date _____ Date: 2021/5/8

17088

Subject: JPEG study

JPEG

The first four bytes of a JPEG file are always ffd8ffe0. This is followed by 2 bytes of header length, and the string "JFIF". These are followed by units used for resolution, horizontal and vertical resolutions.

The RGB color can be translated to the YCbCr (YUV) scheme, using the formula

Y=0.299R+0.587G+0.114B
Cb=0.1687R-0.3313G+0.5B
Cr=0.5R-0.4187G-0.0813B

Subsampling is used in JPEG, so that one Cb and Cr sample is used for each four Y samples, and this saves 50% of the space. Then, discrete cosine transform (DCT) is applied separately to 8 by 8 blocks of data for each YCbCr component. An 8 by 8 table of quantization factors is used. The DCT elements are divided by the corresponding quantization factors and rounded to integers.

The DC components (the (0,0) element of DCT) are replaced by the differences from the DC components of the previous blocks. This differential value may be between -32767 and 32768. These are then grouped into 16 categories, category n being the set of numbers between 2(n-1) and 2^n-1 and their negative counterparts. Each category has its Huffman code, followed by n bits specifying the element in the category. The rest of the DCT elements are called AC elements and are aligned in a zigzag fashion. Each nonzero AC element is combined with the number of zeros preceding it in the alignment and transmitted using a Huffman code.

if jpg 轉成 Bmp 後, 再取 BMP 的

RGB → 轉成 YUV

=> test pitcarn is generated

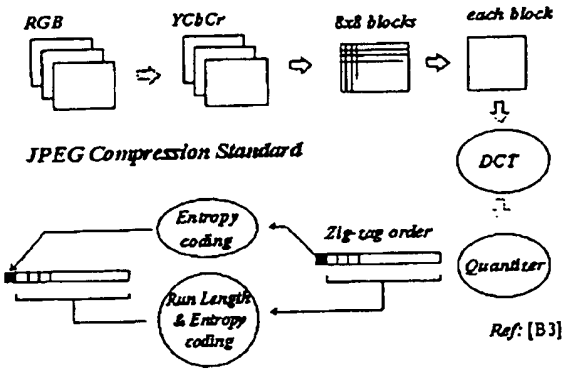
Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2001/5/17

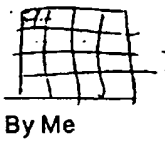
數位影像若不經過壓縮，是佔空間的。通常來說，數位影像可以壓縮到原本的十幾倍都不會看得出來有什麼不一樣。影像壓縮最主要的技術，是利用人眼睛對高頻率信號的不敏感，故將那些信號丟棄或用很少的空間去存一個大概的值。

目前影像壓縮的技術大概以 JPEG 最爲成熟，以下就 JPEG 的大概流程加以說明，若希望知道詳細的 JPEG 壓縮格式，請參看 CCITT Recommendation T.81。



1. 將 RGB 的影像轉換成 Y Cb Cr 三個 Components。
2. 將每一個 Component 切成 8x8 的方塊。
3. 對每一個小方塊做 Discrete Consine Transform (DCT)。
4. 把 DCT 的係數根據 Quantization Table 除以他相對的 Quantization value.
5. 把 DCT 的第一個係數 (DC值) 用 Predictive coding 加上 Entropy coding (可以是用 Huffman coding 或是 Alrithmetic coding) 加以編碼。
6. 把 DCT 其它的係數 (AC值) 用 Run length coding 及 Entropy coding (可以是用 Huffman coding 或是 Alrithmetic coding) 加以編碼。

DCT -> AC . . DC . . -> Huffman



isclosed To And Understood By Me

Name Date Signature: Blair

Name Date Date: 2001/5/9

17090
Subject: huffman code

A quick tutorial on generating a huffman tree

Lets say you have a set of numbers and their frequency of use and want to create a huffman encoding for them:

FREQUENCY	VALUE
5	1
7	2
10	3
15	4
20	5
45	6

- Creating a huffman tree is simple. Sort this list by frequency and make the two-lowest elements into leaves, creating a parent node with a frequency that is the sum of the two lower element's frequencies:

```
      12:*
      /  \
   5:1   7:2
```

The two elements are removed from the list and the new parent node, with frequency 12, is inserted into the list by frequency. So now the list, sorted by frequency, is:

```
10:3
12:*
15:4
20:5
45:6
```

- You then repeat the loop, combining the two lowest elements. This results in:

```
      22:*
      /  \
  10:3   12:*
         /  \
       5:1   7:2
```

and the list is now:

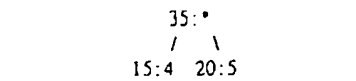
```
15:4
20:5
22:*
45:6
```

You repeat until there is only one element left in the list.

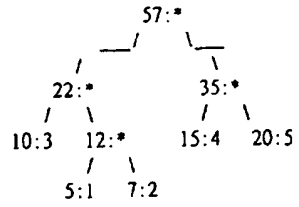
isclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2003/5/19

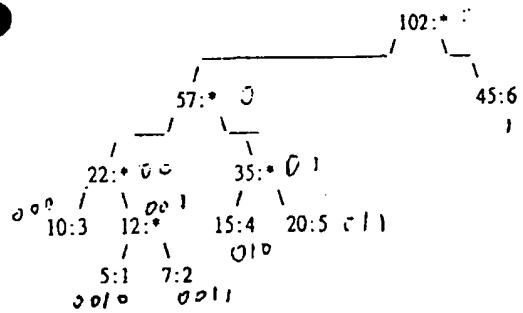
Subject: _____



22:*
35:*
45:6



45:6
57:*



Now the list is just one element containing 102:*, you are done.

This element becomes the root of your binary huffman tree. To generate a huffman code you traverse the tree to the value you want, outputting a 0 every time you take a lefthand branch, and a 1 every time you take a righthand branch. (normally you traverse the tree backwards from the code you want and build the binary huffman encoding string backwards as well, since the *first* bit must start from the top).

Example: The encoding for the value 4 (15:4) is 010. The encoding for the value 6 (45:6) is 1

Decoding a huffman encoding is just as easy : as you read bits in from your input stream you traverse the tree beginning at the root, taking the left hand path if you read a 0 and the right hand path if you read a 1. When you hit a leaf, you have found the code.

Generally, any huffman compression scheme also requires the huffman tree to be written out as part of the file, otherwise the reader cannot decode the data. For a static tree, you don't have to do this since the tree is known and fixed.

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2001/5/9

ubject: _____

The easiest way to output the huffman tree itself is to, starting at the root, dump first the left hand side then the right hand side. For each node you output a 0, for each leaf you output a 1 followed by N bits representing the value. For example, the partial tree in my last example above using 4 bits per value can be represented as follows:

```
000100  fixed 6 bit byte indicates how many bits the value
        for each leaf is stored in.  In this case, 4.
0       root is a node
        left hand side is
10011   a leaf with value 3
        right hand side is
0       another node
        recurse down, left hand side is
10001   a leaf with value 1
        right hand side is
10010   a leaf with value 2
        recursion return
```

So the partial tree can be represented with 000100010011010001l0010, or 23 bits. Not bad!

sclosed To And Understood By Me

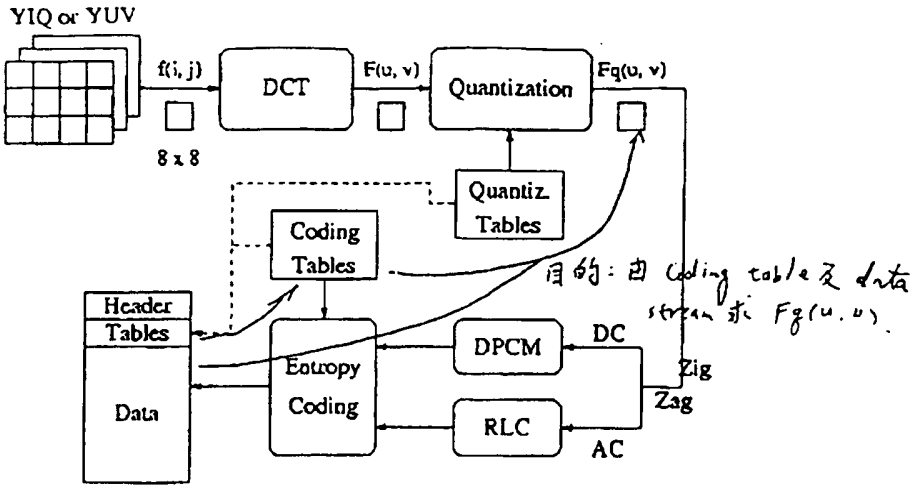
Name	Date
Name	Date

Signature: _____

Date: ٢٠٠٠/٤/١٩

Subject: JPEG Decode step study

17093



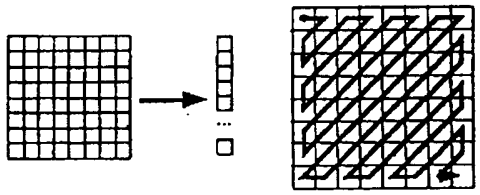
- Decoding -- Reverse the order

4.2.2. Major Steps

- DCT (Discrete Cosine Transformation)
- Quantization
- Zigzag Scan
- DPCM on DC component
- RLE on AC Components
- Entropy Coding

3. Zig-zag Scan

- Why? -- to group low frequency coefficients in top of vector.
- Maps 8 x 8 to a 1 x 64 vector



Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2003/5/10

17004

ubject: _____

4. Differential Pulse Code Modulation (DPCM) on DC component

- DC component is large and varied, but often close to previous value.
- Encode the difference from previous 8 x 8 blocks -- DPCM

5. Run Length Encode (RLE) on AC components

- 1 x 64 vector has lots of zeros in it
- Keeps *skip* and *value*, where *skip* is the number of zeros and *value* is the next non-zero component.
- Send (0,0) as end-of-block sentinel value.

6. Entropy Coding

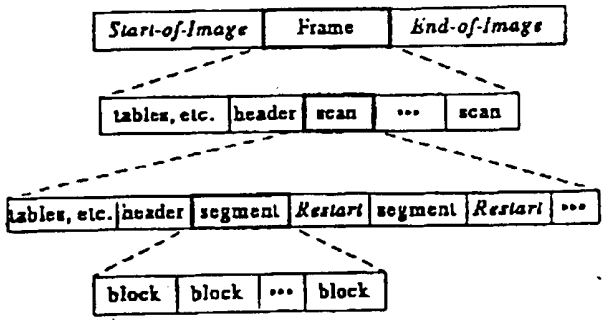
- Categorize DC values into SIZE (number of bits needed to represent) and actual bits.

SIZE	Value
1	-1, 1
2	-3, -2, 2, 3
3	-7..-4, 4..7
4	-15..-8, 8..15
.	.
.	.
.	.
10	-1023..-512, 512..1023

Example: if DC value is 4, 3 bits are needed.

Send off SIZE as Huffman symbol, followed by actual 3 bits.

- For AC components two symbols are used: Symbol_1: (*skip, SIZE*), Symbol_2: actual bits. Symbol_1 (*skip, SIZE*) is encoded using the Huffman coding, Symbol_2 is not encoded.
- Huffman Tables can be custom (sent in header) or default.



- A "Frame" is a picture, a "scan" is a pass through the pixels (e.g., the red component), a "segment"

sclosed To And Understood By Me

Name Date

Name Date

Signature: _____

Date: 11/12/12

is a group of blocks, a "block" is an 8 x 8 group of pixels.

- Frame header:
  sample precision
  (width, height) of image
  number of components
  unique ID (for each component)
  horizontal/vertical sampling factors (for each component)
  quantization table to use (for each component)

- Scan header
  Number of components in scan
  component ID (for each component)
  Huffman table for each component (for each component)

- Misc. (can occur between headers)
  Quantization tables
  Huffman Tables
  Arithmetic Coding Tables
  Comments
  Application Data

4.2.4. Four JPEG Modes

- Sequential Mode
- Lossless Mode
- Progressive Mode
- Hierarchical Mode

** In "Motion JPEG", Sequential JPEG is applied to each image in a video.

1. Sequential Mode

- Each image component is encoded in a single left-to-right, top-to-bottom scan.

Baseline Sequential Mode, the one that we described above, is a simple case of the Sequential mode:
  - It supports only 8-bit images (not 12-bit images)
  - It uses only Huffman coding (not Arithmetic coding)

Disclosed To And Understood By Me

Name	Date	Signature: b/s iv
Name	Date	Date: 2003/5/10

17096

Subject: Huffcomp & Huffdeco

File: D:\Jpegtest\2.txt 2000/5/10, 03:06:54PM

•

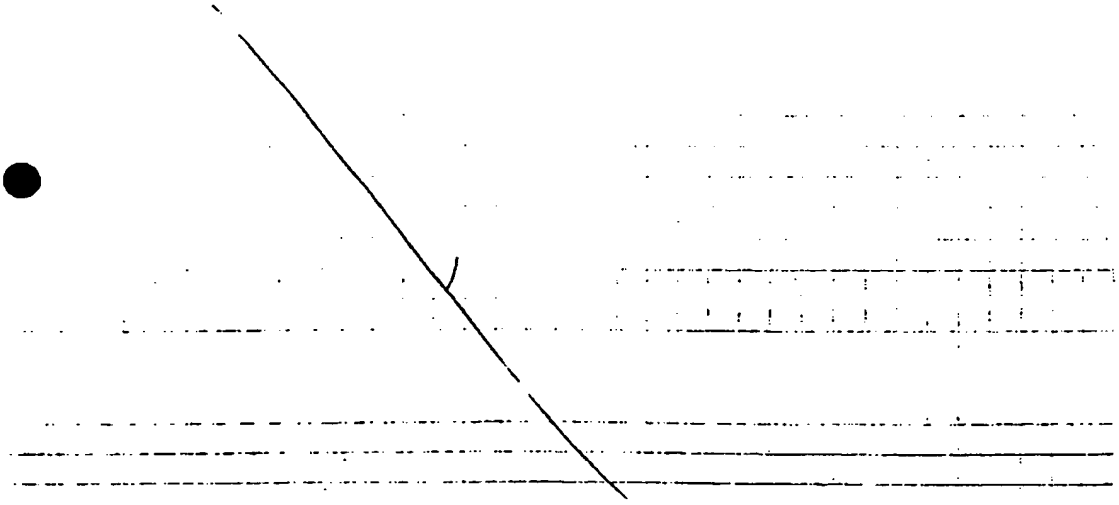
File: D:\Jpegtest\2.txt 2000/5/10, 03:06:54PM

00000000h: 61 0D 0A ; a..

encode

File: D:\Jpegtest\1.txt 2000/5/10, 03:04:32PM

00000000h: 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ; ....
00000010h: 0A 61 80 ; .a

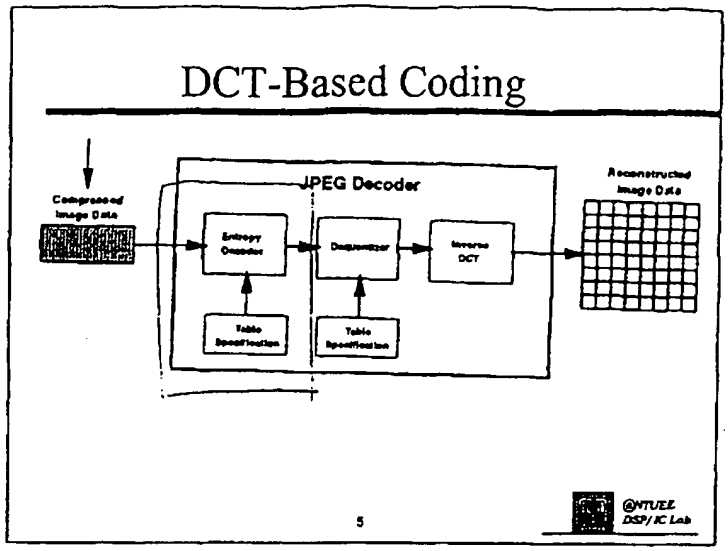


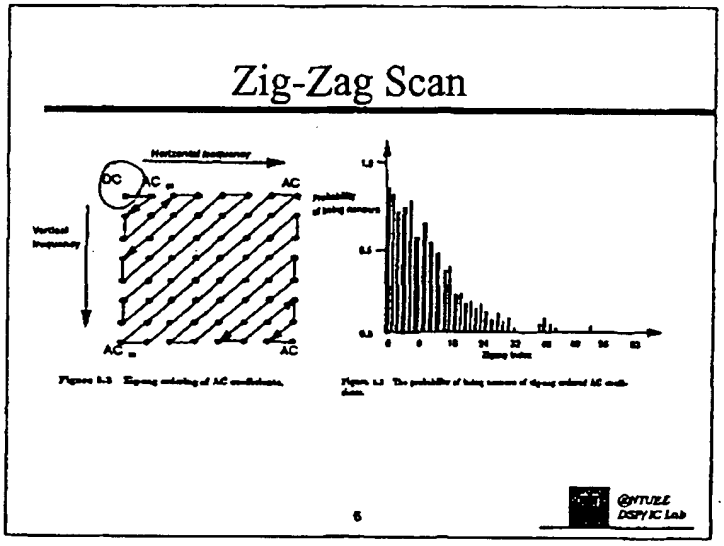
Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/5/10

Subject:

17097





Disclosed To And Understood By Me

Name Date Signature:

Name Date Date:

17093

ubject: _____

Predictive Coding of DC Coefficients

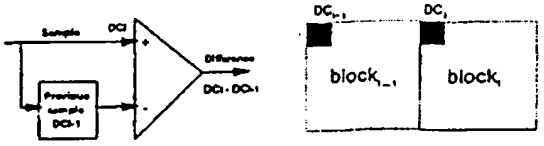


Figure 5.4 Predictive coding of DC coefficients. The difference between the present and the previous DC coefficients is calculated and then coded using JPEG.

Entropy Coding

- Huffman coding and Arithmetic coding
- Huffman encoder (two steps)
  - forming *intermediate symbol sequence*
  - converting *intermediate symbol sequence* into *binary sequence* using Huffman table
- DC : (Size, amplitude),
  - where *size* defines the number of bits required to represent the amplitude, and *amplitude* is the 1's complement amplitude of differential.
  - Only the *size* is huffman coded.
- AC : (runlength, size, amplitude)

closed To And Understood By Me

Name Date Signature: _____

Name Date Date: _____

Subject: Jpeg code study 17099

8 x 8 pixels graph & Black image (R=0 G=0 B=0)

File: D:\Jpegtest\black8x8.ppm 2000/5/10, 04:48:09PM

```
P3
# Created by Paint Shop Pro 6
8 8
255
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0
```

File: D:\Jpegtest\black8x8.jpg 2000/5/10, 04:45:06PM

```
00000000h: FF D8 FF E0 00 10 (4A 46 49 46) 00 01 01 01 (00 48) ;   ??.JFIF.....H
00000010h: (00 48) 00 00
```

JFIF — Y density

File: D:\Jpegtest\b8x8 2000/5/10, 04:46:49PM

```
{ Start Of Image }
{ APP0 Marker
  Length: 16
  Version: 1.1
  Density Unit:  (pixels per inch)
  X Density: 72
  Y Density: 72
  Thumbnail Width: 0
  Thumbnail Height: 0
}
```

length of DQT — index.0 => Y

```
                                 FF DB (00 43) 00 (05 03 04 04 04 03 05 ;
00000020h: 04 04 04 05 05 05 06 07 0C 08 07 07 07 07 0F 0B ;
00000030h: 0B 09 0C 11 0F 12 12 11 0F 11 11 13 16 1C 17 13 ;
00000040h: 14 1A 15 11 11 18 21 1E 1A 1D 1D 1F 1F 1F [illegible] [illegible] ;
00000050h: 22 24 22 1E 24 1C 1E 1F 1E )
```

```
{ Define Quantization Table
  Length:  67
  Table Index: 0
  Table Precision: 0
  Table Values:
        5  3  4  4  4  3  5  4
        4  4  5  5  5  6  7  12
        8  7  7  7  7  15 11 11
        9  12 17 15 18 18 17 15
        17 17 19 22 28 23 19 20
        26 21 17 17 24 33 24 26
        29 29 31 31 31 19 23 34
        36 34 30 36 28 30 31 30
}
```

Disclosed To And Understood By Me

Name	Date	Signature:	blair
Name	Date	Date:	2001/5/15

17100

Subject: ______

```
                                          FF DB (00 43) 00 [illegible] 05 ;    { Define Quantization Table
00000060h: 05 07 06 07 0E 08 08 0E 1E 14 11 14 1E 1E 1E 1E ;       Length: 67
00000070h: 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E ;       Table Index: 1
00000080h: 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E ;       Table Precision: 0
00000090h: 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E 1E)              Table Values:
                                                                        5 5 5 7 6 7 14 8
                                                                        8 14 30 20 17 20 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                        30 30 30 30 30 30 30 30
                                                                    }

   length      height   precision
                                                  FF C0
000000a0h: (00 11) 08 00 08)(00 08)03 01 22 00 02 11 01 03 11 ;  { Start Of Frame
000000b0h: 01                                                      Type: Baseline (Huffman)
              width                                                Length: 17
                                                                   Precision: 8
                                                                   Height: 8
                                                                   Width: 8
                                                                   Component Count: 3
                                                                    Component 1
                                                                     Horizontal Frequency: 2
                                                                     Vertical Frequency: 2
                                                                     Quantization Table: 0
                                                                    Component 2
                                                                     Horizontal Frequency: 1
                                                                     Vertical Frequency: 1
                                                                     Quantization Table: 1
                                                                    Component 3
                                                                     Horizontal Frequency: 1
                                                                     Vertical Frequency: 1
                                                                     Quantization Table: 1
                                                                   }
        length
             FF C4 (00 1F) [illegible] 01 05 01 01 01 01 [illegible] ;
000000c0h: 00 00 00 00 00 00)00 01 02 03 04 05 06 07 08 09 ;
000000d0h: 0A 0B)
        { Define Huffman Table
          Length: 31
           Table Index 0
            Table Class: DC
            Code Counts: 0 1 5 1 1 1 1 1 1 0 0 0 0 0 0 0
            Code Values:  0 1 2 3 4 5 6 7 8 9 a b
        }

                  FF C4 00 B5 10 00 02 01 03 03 02 04 03 05 ;
000000e0h: C5 [illegible] [illegible] [illegible] [illegible] 01 7[illegible] 01 02 03 00 04 11 05 12 21 ;
000000f0h: 31 41 [illegible] 13 51 61 07 22 71 14 32 81 91 A1 08 23 ;
00000100h: 42 B1 C1 15 52 D1 F0 24 33 62 72 82 09 0A 16 17 ;
00000110h: 18 19 1A 25 26 27 28 29 2A 34 35 36 37 38 39 3A ;
00000120h: 43 44 45 46 47 48 49 4A 53 54 55 56 57 58 59 5A ;
00000130h: 63 64 65 66 67 68 69 6A 73 74 75 76 77 78 79 7A ;
00000140h: 83 84 85 86 87 88 89 8A 92 93 94 95 96 97 98 99 ;
00000150h: 9A A2 A3 A4 A5 A6 A7 A8 A9 AA B2 B3 B4 B5 B6 B7 ;
00000160h: B8 B9 BA C2 C3 C4 C5 C6 C7 C8 C9 CA D2 D3 D4 D5 ;
00000170h: D6 D7 D8 D9 DA E1 E2 E3 E4 E5 E6 E7 E8 E9 EA F1 ;
00000180h: F2 F3 F4 F5 F6 F7 F8 F9 FA
       { Define Huffman Table
         Length: 181
          Table Index 0
           Table Class: AC
           Code Counts: 0 2 1 3 3 2 4 3 5 5 4 4 0 0 1 7d
           Code Values:  1  2  3  0  4 11  5 12 21 31 41  6 13 51 61  7
                        22 71 14 32 81 91 a1  8 23 42 b1 c1 15 52 d1 f0
                        24 33 62 72 82  9  a 16 17 18 19 1a 25 26 27 28
                        29 2a 34 35 36 37 38 39 3a 43 44 45 46 47 48 49
                        [illegible]
                        6a 73 74 75 76 77 78 79 7a 83 84 85 86 87 88 89
                        8a 92 93 94 95 96 97 98 99 9a a2 a3 a4 a5 a6 a7
                        a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3 c4 c5
                        c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da e1 e2
                        e3 e4 e5 e6 e7 e8 e9 ea f1 f2 f3 f4 f5 f6 f7 f8
                        f9 fa
       }
```

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: [illegible]

Date: [illegible]

```
                                              FF C4 00 1F 01 00 03 ;
00000190h: 01 01 01 01 01 01 01 01 01 00 00 00 00 00 00 01 ;
000001a0h: 02 03 04 05 06 07 08 09 0A 0B

{ Define Huffman Table
  Length: 31
   Table Index 1
    Table Class: DC
    Code Counts: 0 3 1 1 1 1 1 1 1 1 1 0 0 0 0 0
    Code Values:  0  1  2  3  4  5  6  7  8  9  a  b
}

                                                 FF C4 00 B5 11 00 ;
000001b0h: 02 01 02 04 04 03 04 07 05 04 04 00 01 02 77 00 ;
000001c0h: 01 02 03 11 04 05 21 31 06 12 41 51 07 61 71 13 ;
000001d0h: 22 32 81 08 14 42 91 A1 B1 C1 09 23 33 52 F0 15 ;
000001e0h: 62 72 D1 0A 16 24 34 E1 25 F1 17 18 19 1A 26 27 ;
000001f0h: 28 29 2A 35 36 37 38 39 3A 43 44 45 46 47 48 49 ;
00000200h: 4A 53 54 55 56 57 58 59 5A 63 64 65 66 67 68 69 ;
00000210h: 6A 73 74 75 76 77 78 79 7A 82 83 84 85 86 87 88 ;
00000220h: 89 8A 92 93 94 95 96 97 98 99 9A A2 A3 A4 A5 A6 ;
00000230h: A7 A8 A9 AA B2 B3 B4 B5 B6 B7 B8 B9 BA C2 C3 C4 ;
00000240h: C5 C6 C7 C8 C9 CA D2 D3 D4 D5 D6 D7 D8 D9 DA E2 ;
00000250h: E3 E4 E5 E6 E7 E8 E9 EA F2 F3 F4 F5 F6 F7 F8 F9 ;
00000260h: FA

 { Define Huffman Table

Length: 181
 Table Index 1
  Table Class: AC
  Code Counts: 0 2 1 2 4 4 3 4 7 5 4 4 0 1 2 77
  Code Values:  0  1  2  3 11  4  5 21 31  6 12 41 51  7 61 71
               13 22 32 81  8 14 42 91 a1 b1 c1  9 23 33 52 f0
               15 62 72 d1  a 16 24 34 e1 25 f1 17 18 19 1a 26
               27 28 29 2a 35 36 37 38 39 3a 43 44 45 46 47 48
               49 4a 53 54 55 56 57 58 59 5a 63 64 65 66 67 68
               69 6a 73 74 75 76 77 78 79 7a 82 83 84 85 86 87
               88 89 8a 92 93 94 95 96 97 98 99 9a a2 a3 a4 a5
               a6 a7 a8 a9 aa b2 b3 b4 b5 b6 b7 b8 b9 ba c2 c3
               c4 c5 c6 c7 c8 c9 ca d2 d3 d4 d5 d6 d7 d8 d9 da
               e2 e3 e4 e5 e6 e7 e8 e9 ea f2 f3 f4 f5 f6 f7 f8
               f9 fa
}

                  FF DA 00 0C 03 01 00 02 11 03 11 00 3F 00(F8 ;
00000270h: CA 8A 28 A0 0F)FF D9                              ;
?(?  ?

{ Start Of Scan
  Length:  12
  Scan Count: 3
   Component ID: 1
    AC Entropy Table: 0
    DC Entropy Table: 0
   Component ID: 2
    AC Entropy Table: 1
    DC Entropy Table: 1
   Component ID: 3
    AC Entropy Table: 1
    DC Entropy Table: 1
  Spectral Selection Start: 0
  Spectral Selection End: 63
  Sucessive Approximation High: 0
  Sucessive Approximation Low: 0
}

{ End Of Image }
```

→ 80(b14exter2.jpg)

real data stream: F8 CA 8A 28 A0 0F

和 1×1 及 16×16 除 Height & width 不同, 其余全皆相同

Disclosed To And Understood By Me

Name	Date

Signature: blair

Date: 2000/5/10

17102

Subject:

用 Huff decc 去 decode DC0, DC1, AC0, AC1

=> decode 後之 DC0, DC-1, AC1 = 00 ... AC0 = 01

decode green.jpg 的 DC0, DC1, AC0, AC1

=> decode 後之 DC0 = 01, DC1, AC0, AC1 = 00

Huffman table 有固定之值. ~~[illegible]~~

用 encoder 去 encode bmp => Huffman table 無法得知

Huffman AC code len pies 0 ~ a

DC code 0 ~ b

Disclosed To And Understood By Me

Name Date

Name Date

Signature: [illegible]

Date: 2003/3/11

Subject: Huffman table

17103

B.2.4.2 Huffman table-specification syntax

Figure B.7 specifies the marker segment which defines one or more Huffman table specifications.

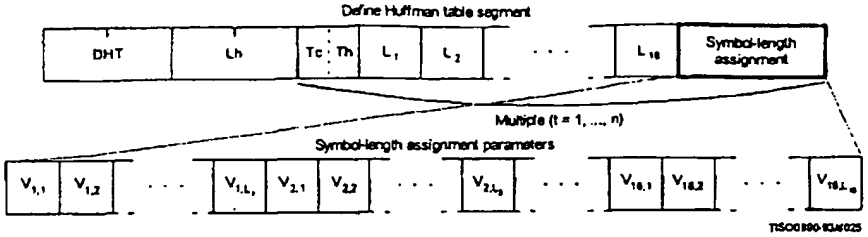


Figure B.7 – Huffman table syntax

The marker and parameters shown in Figure B.7 are defined below. The size and allowed values of each parameter are given in Table B.5.

DHT: Define Huffman table marker – Marks the beginning of Huffman table definition parameters.

Lh: Huffman table definition length – Specifies the length of all Huffman table parameters shown in Figure B.7 (see B.1.1.4).

Tc: Table class – 0 = DC table or lossless table, 1 = AC table.

Th: Huffman table destination identifier – Specifies one of four possible destinations at the decoder into which the Huffman table shall be installed.

L_i: Number of Huffman codes of length i – Specifies the number of Huffman codes for each of the 16 possible lengths allowed by this Specification. L_i's are the elements of the list BITS.

V_{ij}: Value associated with each Huffman code – Specifies, for each i, the value associated with each Huffman code of length i. The meaning of each value is determined by the Huffman coding model. The V_{ij}'s are the elements of the list HUFFVAL.

Table B.5 – Huffman table specification parameter sizes and values

Parameter	Size (bits)	Values			
		Sequential DCT		Progressive DCT	Lossless
		Baseline	Extended		
Lh	16	$2 + \sum_{i=1}^{n} (17 + m_i)$			
Tc	4	0, 1			0
Th	4	0, 1	0-3		
L_i	8	0-255			
$V_{i,j}$	8	0-255			

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2003/5/11

17104
Subject: _____

The value n in Table B.5 is the number of Huffman tables specified in the DHT marker segment. The value m_t is the number of parameters which follow the 16 $L_i(t)$ parameters for Huffman table t, and is given by:

$$m_t = \sum_{i=1}^{16} L_i$$

In general, m_t is different for each table.

Once a Huffman table has been defined for a particular destination, it replaces the previous tables stored in that destination and shall be used when referenced, in the remaining scans of the current image and in subsequent images represented in the abbreviated format for compressed image data. If a table has never been defined for a particular destination, then when this destination is specified in a scan header, the results are unpredictable.

Huffman tables are specified in terms of a 16-byte list (BITS) giving the number of codes for each code length from 1 to 16. This is followed by a list of the 8-bit symbol values (HUFFVAL), each of which is assigned a Huffman code. The symbol values are placed in the list in order of increasing code length. Code lengths greater than 16 bits are not allowed. In addition, the codes shall be generated such that the all-1-bits code word of any length is reserved as a prefix for longer code words.

NOTE – The order of the symbol values within HUFFVAL is determined only by code length. Within a given code length the ordering of the symbol values is arbitrary.

This annex specifies the procedure by which the Huffman tables (of Huffman code words and their corresponding 8-bit symbol values) are derived from the two lists (BITS and HUFFVAL) in the interchange format. However, the way in which these lists are generated is not specified. The lists should be generated in a manner which is consistent with the rules for Huffman coding, and it shall observe the constraints discussed in the previous paragraph. Annex K contains an example of a procedure for generating lists of Huffman code lengths and values which are in accord with these rules.

NOTE – There is no requirement in this Specification that any encoder or decoder shall implement the procedures in precisely the manner specified by the flow charts in this annex. It is necessary only that an encoder or decoder implement the function specified in this annex. The sole criterion for an encoder or decoder to be considered in compliance with this Specification is that it satisfy the requirements given in clause 6 (for encoders) or clause 7 (for decoders), as determined by the compliance tests specified in Part 2.

C.1 Marker segments for Huffman table specification

The DHT marker identifies the start of Huffman table definitions within the compressed image data. B.2.4.2 specifies the syntax for Huffman table specification.

C.2 Conversion of Huffman table specifications to tables of codes and code lengths

Conversion of Huffman table specifications to tables of codes and code lengths uses three procedures. The first procedure (Figure C.1) generates a table of Huffman code sizes. The second procedure (Figure C.2) generates the Huffman codes from the table built in Figure C.1. The third procedure (Figure C.3) generates the Huffman codes in symbol value order.

Given a list BITS (1 to 16) containing the number of codes of each size, and a list HUFFVAL containing the symbol values to be associated with those codes as described above, two tables are generated. The HUFFSIZE table contains a list of code lengths; the HUFFCODE table contains the Huffman codes corresponding to those lengths.

Note that the variable LASTK is set to the index of the last entry in the table.

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date:

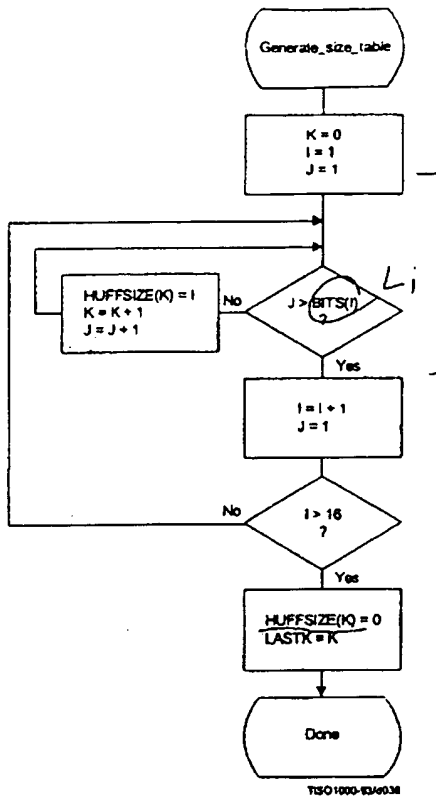


Figure C.1 – Generation of table of Huffman code sizes

Isclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2001/5/11

17166
Subject: _____

A Huffman code table, HUFFCODE, containing a code for each size in HUFFSIZE is generated by the procedure in Figure C.2. The notation "SLL CODE 1" in Figure C.2 indicates a shift-left-logical of CODE by one bit position.

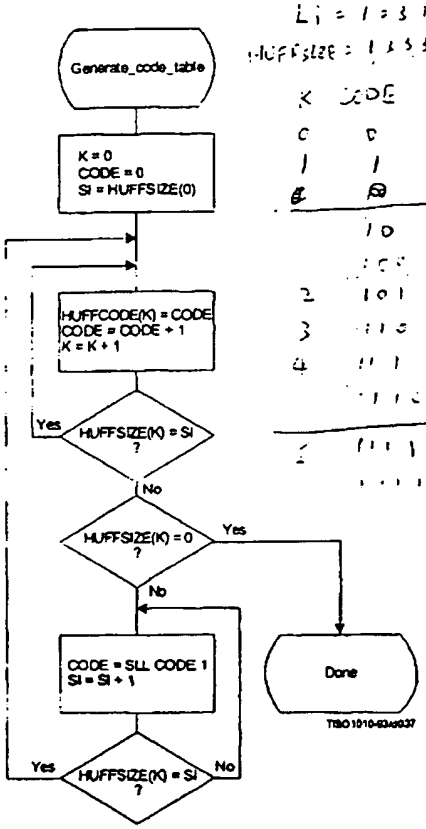


Figure C.2 – Generation of table of Huffman codes

Two tables, HUFFCODE and HUFFSIZE, have now been generated. The entries in the tables are ordered according to increasing Huffman code numeric value and length.

The encoding procedure code tables, EHUFCO and EHUFSI, are created by reordering the codes specified by HUFFCODE and HUFFSIZE according to the symbol values assigned to each code in HUFFVAL.

sclosed To And Understood By Me

Name Date

Name Date

Signature: _____

Date: _____

Figure C.3 illustrates this ordering procedure.

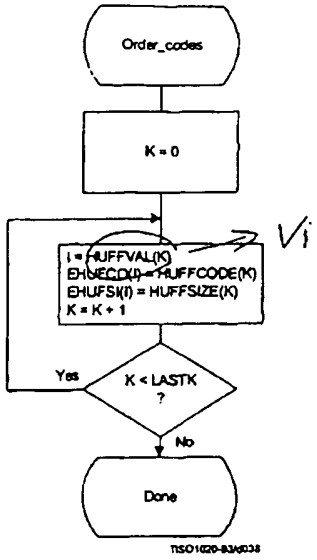


Figure C.3 – Ordering procedure for encoding procedure code tables

C.3 Bit ordering within bytes

The root of a Huffman code is placed toward the MSB (most-significant-bit) of the byte, and successive bits are placed in the direction MSB to LSB (least-significant-bit) of the byte. Remaining bits, if any, go into the next byte following the same rules.

Integers associated with Huffman codes are appended with the MSB adjacent to the LSB of the preceding Huffman code.

E.2.3 Control procedure for decoding a scan

Figure E.8 shows the decoding of a scan.

The loop is terminated when the expected number of restart intervals has been decoded.

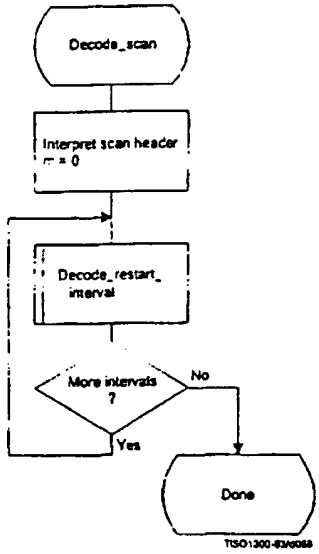


Figure E.8 – Control procedure for decoding a scan

Disclosed To And Understood By Me

Name Date

Name Date

Signature:

Date:

Subject: Huffman encode

F.1.1.5 Encoding models for the sequential DCT procedures

The two dimensional array of quantized DCT coefficients is rearranged in a zig-zag sequence order defined in A.3.6. The zig-zag order coefficients are denoted ZZ (0) through ZZ(63) with:

$$ZZ(0) = Sq_{00}, ZZ(1) = Sq_{01}, ZZ(2) = Sq_{10}, \ldots, ZZ(63) = Sq_{77}$$

Sq_{vu} are defined in Figure A.6.

Two coding procedures are used, one for the DC coefficient ZZ(0) and the other for the AC coefficients ZZ(1)..ZZ(63). The coefficients are encoded in the order in which they occur in zig-zag sequence order, starting with the DC coefficient. The coefficients are represented as two's complement integers.

F.1.1.5.1 Encoding model for DC coefficients

The DC coefficients are coded differentially, using a one-dimensional predictor, PRED, which is the quantized DC value from the most recently coded 8 x 8 block from the same component. The difference, DIFF, is obtained from

$$DIFF = ZZ(0) - PRED$$

At the beginning of the scan and at the beginning of each restart interval, the prediction for the DC coefficient prediction is initialized to 0. (Recall that the input data have been level shifted to two's complement representation.)

F.1.1.5.2 Encoding model for AC coefficients

Since many coefficients are zero, runs of zeros are identified and coded efficiently. In addition, if the remaining coefficients in the zig-zag sequence order are all zero, this is coded explicitly as an end-of-block (EOB).

F.1.2 Baseline Huffman encoding procedures

The baseline encoding procedure is for 8-bit sample precision. The encoder may employ up to two DC and two AC Huffman tables within one scan.

F.1.2.1 Huffman encoding of DC coefficients

F.1.2.1.1 Structure of DC code table

The DC code table consists of a set of Huffman codes (maximum length 16 bits) and appended additional bits (in most cases) which can code any possible value of DIFF, the difference between the current DC coefficient and the prediction. The Huffman codes for the difference categories are generated in such a way that no code consists entirely of 1-bits (X'FF' prefix marker code avoided).

The two's complement difference magnitudes are grouped into 12 categories, SSSS, and a Huffman code is created for each of the 12 difference magnitude categories (see Table F.1).

For each category, except SSSS = 0, an additional bits field is appended to the code word to uniquely identify which difference in that category actually occurred. The number of extra bits is given by SSSS; the extra bits are appended to the LSB of the preceding Huffman code, most significant bit first. When DIFF is positive, the SSSS low order bits of DIFF are appended. When DIFF is negative, the SSSS low order bits of (DIFF – 1) are appended. Note that the most significant bit of the appended bit sequence is 0 for negative differences and 1 for positive differences.

F.1.2.1.2 Defining Huffman tables for the DC coefficients

The syntax for specifying the Huffman tables is given in Annex B. The procedure for creating a code table from this information is described in Annex C. No more than two Huffman tables may be defined for coding of DC coefficients. Two examples of Huffman tables for coding of DC coefficients are provided in Annex K.

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2001/5/11

17112

Subject: _____

Table F.1 – Difference magnitude categories for DC coding

SSSS	DIFF values
0	0
1	-1,1
2	-3,-2,2,3
3	-7..-4,4..7
4	-15..-8,8..15
5	-31..-16,16..31
6	-63..-32,32..63
7	-127..-64,64..127
8	-255..-128,128..255
9	-511..-256,256..511
10	-1 023..-512,512..1 023
11	-2 047..-1 024,1 024..2 047

F.1.2.1.3 **Huffman encoding procedures for DC coefficients**

The encoding procedure is defined in terms of a set of extended tables, XHUFCO and XHUFSI, which contain the complete set of Huffman codes and sizes for all possible difference values. For full 12-bit precision the tables are relatively large. For the baseline system, however, the precision of the differences may be small enough to make this description practical.

XHUFCO and XHUFSI are generated from the encoder tables EHUFCO and EHUFSI (see Annex C) by appending to the Huffman codes for each difference category the additional bits that completely define the difference. By definition, XHUFCO and XHUFSI have entries for each possible difference value. XHUFCO contains the concatenated bit pattern of the Huffman code and the additional bits field; XHUFSI contains the total length in bits of this concatenated bit pattern. Both are indexed by DIFF, the difference between the DC coefficient and the prediction.

The Huffman encoding procedure for the DC difference, DIFF, is:

SIZE = XHUFSI(DIFF)

CODE = XHUFCO(DIFF)

code SIZE bits of CODE

where DC is the quantized DC coefficient value and PRED is the predicted quantized DC value. The Huffman code (CODE) (including any additional bits) is obtained from XHUFCO and SIZE (length of the code including additional bits) is obtained from XHUFSI, using DIFF as the index to the two tables.

F.1.2.2 **Huffman encoding of AC coefficients**

F.1.2.2.1 **Structure of AC code table**

Each non-zero AC coefficient in ZZ is described by a composite 8-bit value, RS, of the form

RS = binary 'RRRRSSSS'

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2009/5/11

The 4 least significant bits, 'SSSS', define a category for the amplitude of the next non-zero coefficient in ZZ, and the 4 most significant bits, 'RRRR', give the position of the coefficient in ZZ relative to the previous non-zero coefficient (i.e. the run-length of zero coefficients between non-zero coefficients). Since the run length of zero coefficients may exceed 15, the value 'RRRRSSSS' = X'F0' is defined to represent a run length of 15 zero coefficients followed by a coefficient of zero amplitude. (This can be interpreted as a run length of 16 zero coefficients.) In addition, a special value 'RRRRSSSS' = '00000000' is used to code the end-of-block (EOB), when all remaining coefficients in the block are zero.

The general structure of the code table is illustrated in Figure F.1. The entries marked "N/A" are undefined for the baseline procedure.

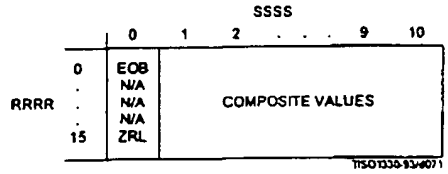


Figure F.1 – Two-dimensional value array for Huffman coding

The magnitude ranges assigned to each value of SSSS are defined in Table F.2.

Table F.2 – Categories assigned to coefficient values

SSSS	AC coefficients
1	-1,1
2	-3,-2,2,3
3	-7..-4,4..7
4	-15..-8,8..15
5	-31..-16,16..31
6	-63..-32,32..63
7	-127..-64,64..127
8	-255..-128,128..255
9	-511..-256,256..511
10	-1 023..-512,512..1 023

The composite value, RRRRSSSS, is Huffman coded and each Huffman code is followed by additional bits which specify the sign and exact amplitude of the coefficient.

The AC code table consists of one Huffman code (maximum length 16 bits, not including additional bits) for each possible composite value. The Huffman codes for the 8-bit composite values are generated in such a way that no code consists entirely of 1-bits.

Disclosed To And Understood By Me

Name Date Signature: Blair

Name Date Date: 2/19/5/11

The format for the additional bits is the same as in the coding of the DC coefficients. The value of SSSS gives the number of additional bits required to specify the sign and precise amplitude of the coefficient. The additional bits are either the low-order SSSS bits of ZZ(K) when ZZ(K) is positive or the low-order SSSS bits of ZZ(K) – 1 when ZZ(K) is negative. ZZ(K) is the *K*th coefficient in the zig-zag sequence of coefficients being coded.

F.1.2.2.2 Defining Huffman tables for the AC coefficients

The syntax for specifying the Huffman tables is given in Annex B. The procedure for creating a code table from this information is described in Annex C.

In the baseline system no more than two Huffman tables may be defined for coding of AC coefficients. Two examples of Huffman tables for coding of AC coefficients are provided in Annex K.

F.1.2.2.3 Huffman encoding procedures for AC coefficients

As defined in Annex C, the Huffman code table is assumed to be available as a pair of tables, EHUFCO (containing the code bits) and EHUFSI (containing the length of each code in bits), both indexed by the composite value defined above.

The procedure for encoding the AC coefficients in a block is shown in Figures F.2 and F.3. In Figure F.2, K is the index to the zig-zag scan position and R is the run length of zero coefficients.

The procedure "Append EHUFSI(X'F0') bits of EHUFCO(X'F0')" codes a run of 16 zero coefficients (ZRL code of Figure F.1). The procedure "Code EHUFSI(0) bits of EHUFCO(0)" codes the end-of-block (EOB code). If the last coefficient (K = 63) is not zero, the EOB code is bypassed.

CSIZE is a procedure which maps an AC coefficient to the SSSS value as defined in Table F.2.

F.1.2.3 Byte stuffing

In order to provide code space for marker codes which can be located in the compressed image data without decoding, byte stuffing is used.

Whenever, in the course of normal encoding, the byte value X'FF' is created in the code string, a X'00' byte is stuffed into the code string.

If a X'00' byte is detected after a X'FF' byte, the decoder must discard it. If the byte is not zero, a marker has been detected, and shall be interpreted to the extent needed to complete the decoding of the scan.

Byte alignment of markers is achieved by padding incomplete bytes with 1-bits. If padding with 1-bits creates a X'FF' value, a zero byte is stuffed before adding the marker.

F.1.3 Extended sequential DCT-based Huffman encoding process for 8-bit sample precision

This process is identical to the Baseline encoding process described in F.1.2, with the exception that the number of sets of Huffman table destinations which may be used within the same scan is increased to four. Four DC and four AC Huffman table destinations is the maximum allowed by this Specification.

F.1.4 Extended sequential DCT-based arithmetic encoding process for 8-bit sample precision

This subclause describes the use of arithmetic coding procedures in the sequential DCT-based encoding process.

NOTE – The arithmetic coding procedures in this Specification are defined for the maximum precision to encourage interchangeability.

The arithmetic coding extensions have the same DCT model as the Baseline DCT encoder. Therefore, Annex F.1.1 also applies to arithmetic coding. As with the Huffman coding technique, the binary arithmetic coding technique is lossless. It is possible to transcode between the two systems without either FDCT or IDCT computations, and without modification of the reconstructed image.

The basic principles of adaptive binary arithmetic coding are described in Annex D. Up to four DC and four AC conditioning table destinations and associated statistics areas may be used within one scan.

The arithmetic encoding procedures for encoding binary decisions, initializing the statistics area, initializing the encoder, terminating the code string, and adding restart markers are listed in Table D.1 of Annex D.

closed To And Understood By Me

Name Date Signature:

Name Date Date:

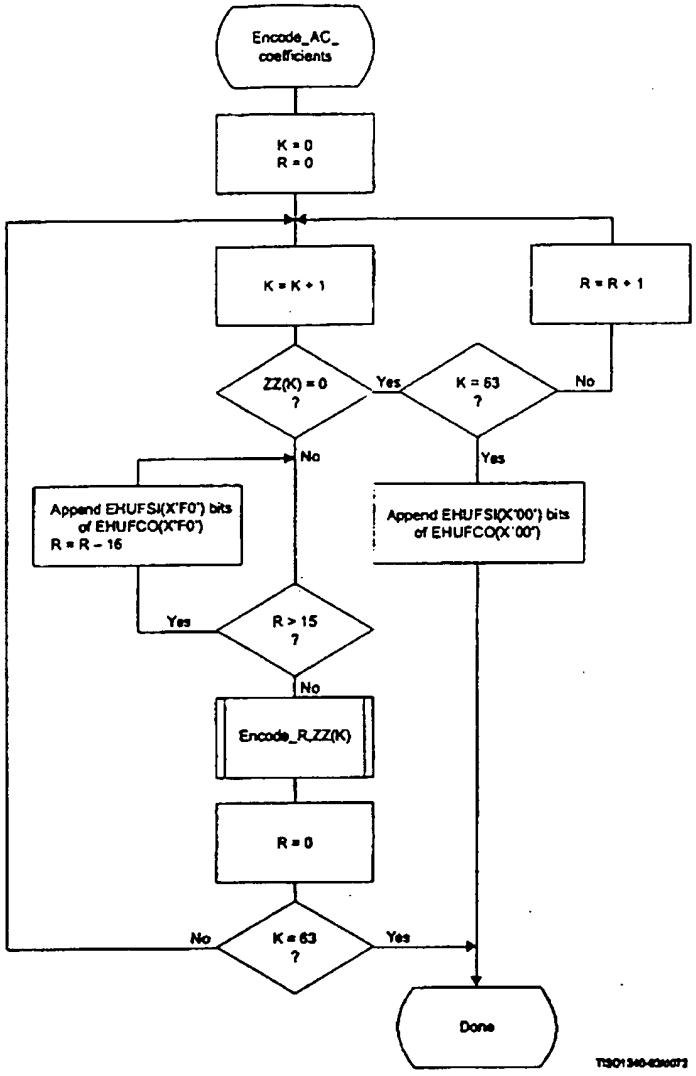


Figure F.2 – Procedure for sequential encoding of AC coefficients with Huffman coding

Disclosed To And Understood By Me

Name Date

Name Date

Signature: _____

Date: 2001/5/11

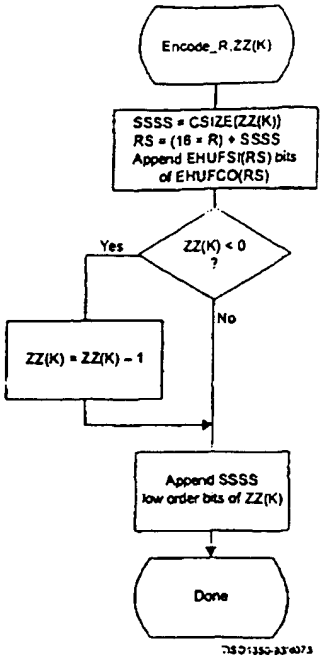


Figure F.3 – Sequential encoding of a non-zero AC coefficient

Some of the procedures in Table D.1 are used in the higher level control structure for scans and restart intervals described in Annex E. At the beginning of scans and restart intervals, the probability estimates used in the arithmetic coder are reset to the standard initial value as part of the Initenc procedure which restarts the arithmetic coder. At the end of scans and restart intervals, the Flush procedure is invoked to empty the code register before the next marker is appended.

F.1.4.1 Arithmetic encoding of DC coefficients

The basic structure of the decision sequence for encoding a DC difference value, DIFF, is shown in Figure F.4.

The context-index S0 and other context-indices used in the DC coding procedures are defined in Table F.4 (see F.1.4.4.1.3). A 0-decision is coded if the difference value is zero and a 1-decision is coded if the difference is not zero. If the difference is not zero, the sign and magnitude are coded using the procedure Encode_V(S0), which is described in F.1.4.3.1.

F.1.4.2 Arithmetic encoding of AC coefficients

The AC coefficients are coded in the order in which they occur in the zig-zag sequence ZZ(1,...,63). An end-of-block (EOB) binary decision is coded before coding the first AC coefficient in ZZ, and after each non-zero coefficient. If the EOB occurs, all remaining coefficients in ZZ are zero. Figure F.5 illustrates the decision sequence. The equivalent procedure for the Huffman coder is found in Figure F.2.

Disclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 7/22/11

Subject: _____ 17117

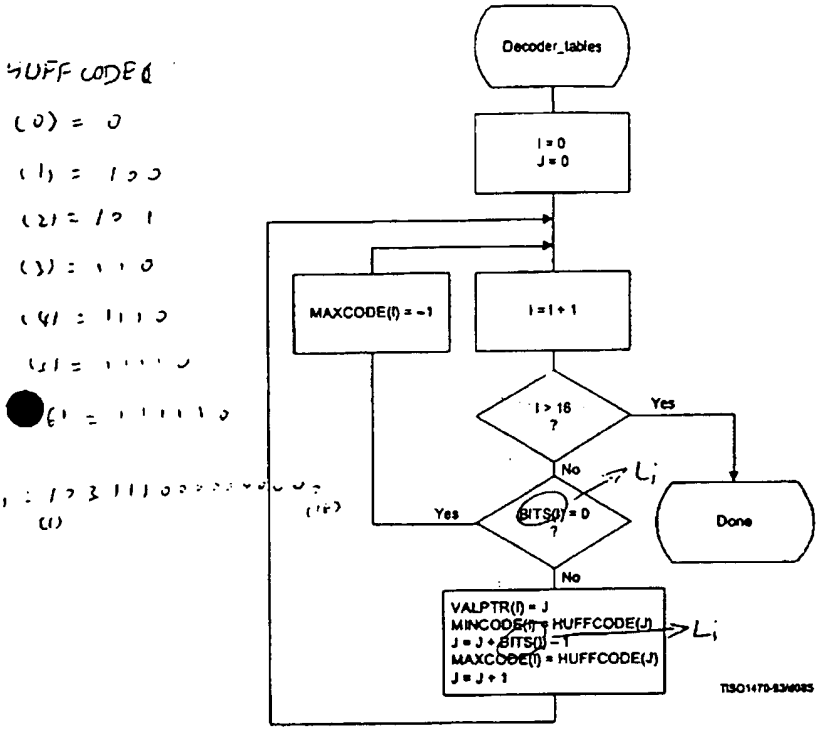


Figure F.15 – Decoder table generation

HUFF CODE

(0) = 0

(1) = 1 0 0

(2) = 1 0 1

(3) = 1 1 0

(4) = 1 1 1 0

(5) = 1 1 1 1 0

(6) = 1 1 1 1 1 0

I J MINCODE MAXCODE VALPTR

0 0 (1) = 0 (1) = 0

1 1

2 (2) = -1

3 3 (3) = 100 (3) = 110 (3) = 1

4 4

4 4 (4) = 1110 (4) = 4

5 5

5 (5) = 5

Disclosed To And Understood By Me

Name Date Signature: _____

Name Date Date: _____

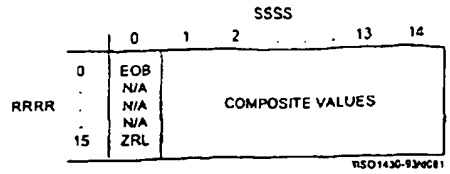


Figure F.11 – Two-dimensional value array for Huffman coding

Table F.7 – Values assigned to coefficient amplitude ranges

SSSS	AC coefficients
11	-2 047..-1 024,1 024..2 047
12	-4 095..-2 048,2 048..4 095
13	-8 191..-4 096,4 096..8 191
14	-16 383..-8 192,8 192..16 383

F.1.6 Extended sequential DCT-based arithmetic encoding process for 12-bit sample precision

The process is identical to the sequential DCT process for 8-bit precision except for changes in the precision of the FDCT computation.

The structure of the encoding procedure is identical to that specified in F.1.4 which was already defined for a 12-bit sample precision.

F.2 Sequential DCT-based decoding processes

F.2.1 Sequential DCT-based control procedures and coding models

F.2.1.1 Control procedures for sequential DCT-based decoders

The control procedures for decoding compressed image data and its constituent parts – the frame, scan, restart interval and MCU – are given in Figures E.6 to E.10. The procedure for decoding a MCU (Figure E.10) repetitively calls the procedure for decoding a data unit. For DCT-based decoders the data unit is an 8 × 8 block of samples.

F.2.1.2 Procedure for decoding an 8 × 8 block data unit

In the sequential DCT-based decoding process, decoding an 8 × 8 block data unit consists of the following procedures:

a) decode DC coefficient for 8 × 8 block using the DC table destination specified in the scan header;

b) decode AC coefficients for 8 × 8 block using the AC table destination specified in the scan header;

c) dequantize using table destination specified in the frame header and calculate the inverse 8 × 8 DCT.

F.2.1.3 Decoding models for the sequential DCT procedures

Two decoding procedures are used, one for the DC coefficient ZZ(0) and the other for the AC coefficients ZZ(1)..ZZ(63). The coefficients are decoded in the order in which they occur in the zig-zag sequence order, starting with the DC coefficient. The coefficients are represented as two's complement integers.

Disclosed To And Understood By Me

Name Date Signature:

Name Date Date:

F.2.1.3.1 Decoding model for DC coefficients

The decoded difference, DIFF, is added to PRED, the DC value from the most recently decoded 8 × 8 block from the same component. Thus ZZ(0) = PRED + DIFF.

At the beginning of the scan and at the beginning of each restart interval, the prediction for the DC coefficient is initialized to zero.

F.2.1.3.2 Decoding model for AC coefficients

The AC coefficients are decoded in the order in which they occur in ZZ. When the EOB is decoded, all remaining coefficients in ZZ are initialized to zero.

F.2.1.4 Dequantization of the quantized DCT coefficients

The dequantization of the quantized DCT coefficients as described in Annex A, is accomplished by multiplying each quantized coefficient value by the quantization table value for that coefficient. The decoder shall be able to use up to four quantization table destinations.

F.2.1.5 Inverse DCT (IDCT)

The mathematical definition of the IDCT is given in A.3.3.

After computation of the IDCT, the signed output samples are level-shifted, as described in Annex A, converting the output to an unsigned representation. For 8-bit precision the level shift is performed by adding 128. For 12-bit precision the level shift is performed by adding 2 048. If necessary, the output samples shall be clamped to stay within the range appropriate for the precision (0 to 255 for 8-bit precision and 0 to 4 095 for 12-bit precision).

F.2.2 Baseline Huffman Decoding procedures

The baseline decoding procedure is for 8-bit sample precision. The decoder shall be capable of using up to two DC and two AC Huffman tables within one scan.

F.2.2.1 Huffman decoding of DC coefficients

The decoding procedure for the DC difference, DIFF, is:

T = DECODE

DIFF = RECEIVE(T)

DIFF = EXTEND(DIFF,T)

where DECODE is a procedure which returns the 8-bit value associated with the next Huffman code in the compressed image data (see F.2.2.3) and RECEIVE(T) is a procedure which places the next T bits of the serial bit string into the low order bits of DIFF, MSB first. If T is zero, DIFF is set to zero. EXTEND is a procedure which converts the partially decoded DIFF value of precision T to the full precision difference. EXTEND is shown in Figure F.12.

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2007/5/12

17120

Subject:

```
//
//  De...      ...on:
//
//                 ...al.
//
//                 ...ct.    extends the sign bit of a decoded value.
//
//  P...           ...:
//                      ...le
//                 ...   of the bit value
//

static         in... ...tend (int vv, int tt)
{
  // ...            ...efined in Section F.2.2.1 Figure F.12
   //               ... vv is the sign bit. One is for
   // ...           ..., and zero is for negative values.
   int          ... <<    ... - 1) ;
   if  ...      ...)
   {
                          - 1 ;
                      ;
   }
   else
   {
   }
}
```

ISO/IEC 10918-1 : 1993(E)

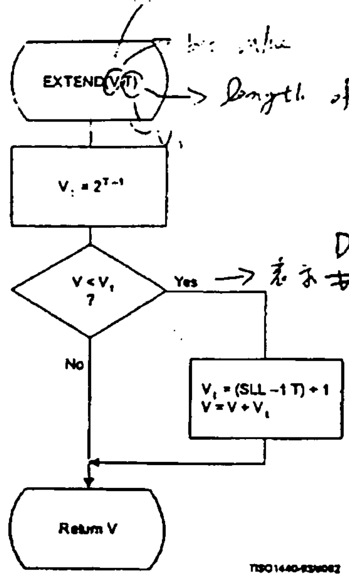


Figure F.12 – Extending the sign bit of a decoded value in V

if V = 010
=> Vt = 100 > V
=> Vt = -1000 + 1
=> V = 010 - 1000 + 1
= 011 - 1000
= -101

again V = 010
=> Vt = 100 > V

=> V = 11011

F.2.2.2 Decoding procedure for AC coefficients

The decoding procedure for AC coefficients is shown in Figures F.13 and F.14.

sclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2011/5/12

Subject:

```
//
//   :
//
//                    :    ec :    ) data unit in a sequential scan.
//
//   P.              :
//                         od    that owns this component
//                 .         Th  row and column for this data unit.

//
void JpegDecoderComponent::DecodeSequential (JpegDecoder &decoder,
                                             unsigned int mcurow,
                                             unsigned int mcuc
{
  JpegDecoderCoefficientBlock data ;
  memset (&data, 0, sizeof (data)) ;

  // Decode the DC differce value.
  // Section F.2.2.1
  unsigned int count ; // called T in F.2.2.1
  count = dc_table->Decode (decoder) ;
  int bits = decoder.Receive (count) ;
  int diff = Extend (bits, count) ;

  // Create the DC value from the difference and the previous DC value.
  int dc = diff + last_dc_value ;
  last_dc_value = dc ;
  data [0][0] = dc ;

  // Decode the AC coefficients.
  // Section F.2.2.2 Figure F.13
  for (unsigned int kk = 1 ; kk < JpegSampleSize ; ++ kk)
  {
    UBYTE2 rs = ac_table->Decode (decoder) ;
    UBYTE2 ssss = (UBYTE2) (rs & 0xF) ;
    UBYTE2 rrrr = (UBYTE2) (rs >> 0x4) ;

    if (ssss == 0)
    {
      // ssss is zero then rrrr should either be 15 or zero according
      // Figure F.1. 0 means that the rest of the coefficients are ze
      // while 15 means the next 16 coefficients are zero. We are not       ing
      // for other values because Figure F.13 shows values other than
      // as being treated as zero.
      if (rrrr != 15)
        break ;
      kk += 15 ; // Actually 16 since one more gets added by the loop
    }
    else
    {
      // If ssss is non-zero then rrrr gives the number of zero coeff
      // to skip.

      kk += rrrr ;
      if (kk >= JpegSampleSize)
        throw EJpegFatal ("Value out of range") ;

      // Receive and extend the additional bits.
      // Section F2.2.2 Figure F.14
      int bits = decoder.Receive (ssss) ;
      int value = Extend (bits, ssss) ;
      (&data [0][0])[JpegZigZagInputOrder (kk)] = value ;
    }
  }
  data_units [mcurow * du_cols + mcucol].InverseDCT (data,
                                              *quantization_ta
  return ;
}
```

Disclosed To And Understood By Me

Name Date

Name Date

Signature: b kir

Date: 2"/8/12

Subject: 17122

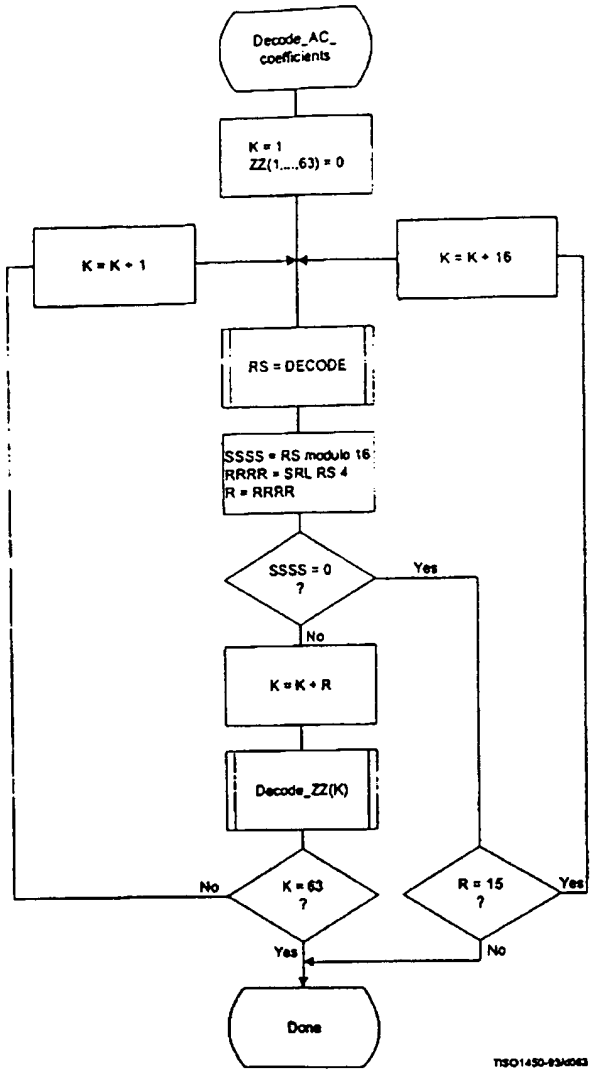


Figure F.13 – Huffman decoding procedure for AC coefficients

sclosed To And Understood By Me

Name Date

Name Date

Signature: [signature]

Date: [date]

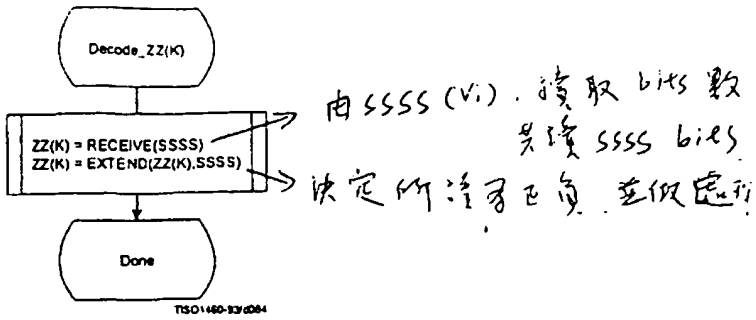


Figure F.14 – Decoding a non-zero AC coefficient

The decoding of the amplitude and sign of the non-zero coefficient is done in the procedure "Decode_ZZ(K)", shown in Figure F.14.

DECODE is a procedure which returns the value, RS, associated with the next Huffman code in the code stream (see F.2.2.3). The values SSSS and R are derived from RS. The value of SSSS is the four low order bits of the composite value and R contains the value of RRRR (the four high order bits of the composite value). The interpretation of these values is described in F.1.2.2. EXTEND is shown in Figure F.12.

F.2.2.3 The DECODE procedure

The DECODE procedure decodes an 8-bit value which, for the DC coefficient, determines the difference magnitude category. For the AC coefficient this 8-bit value determines the zero run length and non-zero coefficient category.

Three tables, HUFFVAL, HUFFCODE, and HUFFSIZE, have been defined in Annex C. This particular implementation of DECODE makes use of the ordering of the Huffman codes in HUFFCODE according to both value and code size. Many other implementations of DECODE are possible.

NOTE – The values in HUFFVAL are assigned to each code in HUFFCODE and HUFFSIZE in sequence. There are no ordering requirements for the values in HUFFVAL which have assigned codes of the same length.

The implementation of DECODE described in this subclause uses three tables, MINCODE, MAXCODE and VALPTR, to decode a pointer to the HUFFVAL table. MINCODE, MAXCODE and VALPTR each have 16 entries, one for each possible code size. MINCODE(I) contains the smallest code value for a given length I. MAXCODE(I) contains the largest code value for a given length I, and VALPTR(I) contains the index to the start of the list of values in HUFFVAL which are decoded by code words of length I. The values in MINCODE and MAXCODE are signed 16-bit integers: therefore, a value of –1 sets all of the bits.

The procedure for generating these tables is shown in Figure F.15. The procedure for DECODE is shown in Figure F.16. Note that the 8-bit "VALUE" is returned to the procedure which invokes DECODE.

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2022/5/12

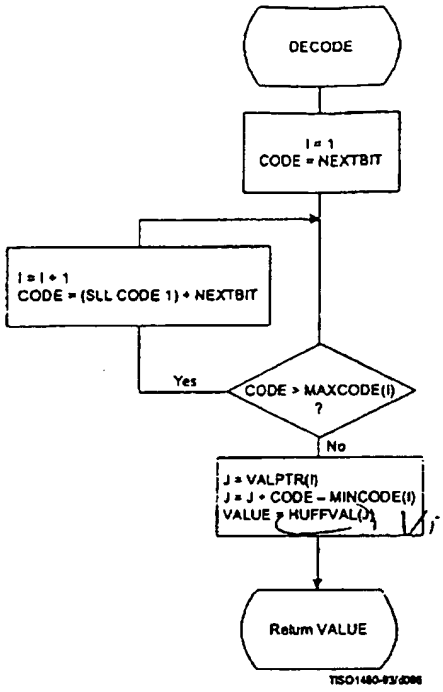


Figure F.16 – Procedure for DECODE

T = DECODE

由 Bitstream → 找出对应之 V_i

=> T = DECODE = V_i

sclosed To And Understood By Me

Name	Date
Name	Date

Signature: olair

Date: 2005/5/12

Subject: _____

F.2.2.4 The RECEIVE procedure

RECEIVE(SSSS) is a procedure which places the next SSSS bits of the entropy-coded segment into the low order bits of DIFF, MSB first. It calls NEXTBIT and it returns the value of DIFF to the calling procedure (see Figure F.17).

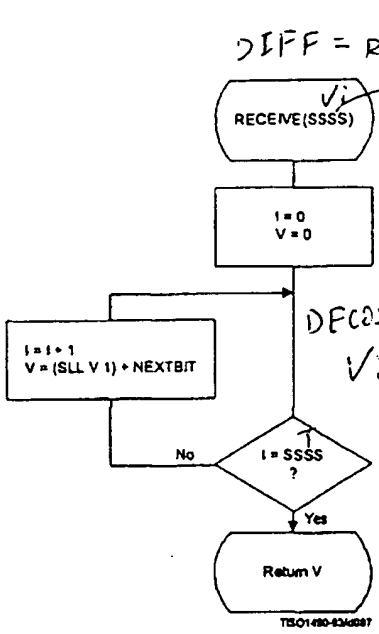


Figure F.17 – Procedure for RECEIVE(SSSS)

F.2.2.5 The NEXTBIT procedure

NEXTBIT reads the next bit of compressed data and passes it to higher level routines. It also intercepts and removes stuff bytes and detects markers. NEXTBIT reads the bits of a byte starting with the MSB (see Figure F.18).

Before starting the decoding of a scan, and after processing a RST marker, CNT is cleared. The compressed data are read one byte at a time, using the procedure NEXTBYTE. Each time a byte, B, is read, CNT is set to 8.

The only valid marker which may occur within the Huffman coded data is the RST_m marker. Other than the EOI or markers which may occur at or before the start of a scan, the only marker which can occur at the end of the scan is the DNL (define-number-of-lines).

Normally, the decoder will terminate the decoding at the end of the final restart interval before the terminating marker is intercepted. If the DNL marker is encountered, the current line count is set to the value specified by that marker. Since the DNL marker can only be used at the end of the first scan, the scan decode procedure must be terminated when it is encountered.

isclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2001/5/12

Subject:

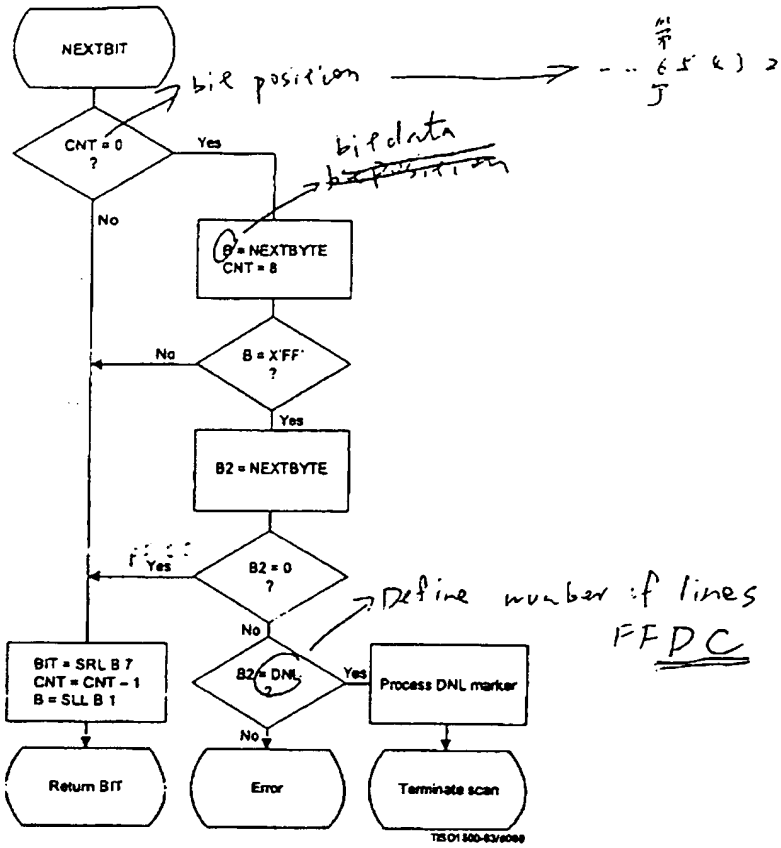


Figure F.18 – Procedure for fetching the next bit of compressed data

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date:

Subject: _____

File: D:\Install\Compressed Image File Formats\Examples\c08\src\jpegdeco.cpp 19
98/12/22, 08:52:22PM

```
//
//  Description:
//
//    This function returns the next raw bit in the input stream.
//
//    Bits are read from high order to low.
//
//  Return Value:
//
//    The next bit (0, 1)
//

// This function returns the next bit in the input stream.
int JpegDecoder::NextBit ()
{
  // Section F.2.2.5 Figure F.18.
  // CNT is called bitposition
  // B is called bitdata
  if (bit_position == 0)
  {
    // We are out of data so read the next byte from the input stream.
    input_stream->read ((char *) &bit_data, sizeof (bit_data)) ;
    if (input_stream->eof ())
      throw EJpegBadData ("Premature end of file") ;
    // Reset the bit read position starting with the highest order bit. We
    // read high to low.
    bit_position = CHAR_BIT * sizeof (bit_data) ;
    if (bit_data == 0xFF)
    {
      // 0xFF could start a marker. The sequence 0xFF, 0x00 is used to
      // to represent the value 0xFF. The only other marker that is legal
      // at this point is a DNL marker.
      UBYTE1 b2 ;
      input_stream->read ((char *) &b2, 1) ;
      if (input_stream->eof ())
        throw EJpegBadData ("Premature end of file") ;
      if (b2 != 0)
      {
        if (b2 == DNL)
        {
          // DNL markers should not occur within the supported frame types.
          throw EJpegBadData ("Unexpected Marker DNL") ;
        }
        else
        {
          throw EJpegBadData ("Unexpected Marker") ;
        }
      }
    }
  }

  // Consume one bit of the input.
  --bit_position ;
  // Shift the value to the low order bit position.
  UBYTE1 result = (UBYTE1) ((bit_data >> bit_position) & 1) ;
  return result ;
}
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/5/12

17128

ubject: ...

```
//
//  Description:
//
//    This function reads a Huffman table from the input stream. A DHT
//    marker can define more than one Huffman table. This function reads
//    just one of those tables.
//
//  Parameters:
//    decoder:  The JPEG decoder that owns the Huffman table
//
//  Return Value:
//    The size of the Huffman table in the input stream
//    (the number of bytes read).
//
unsigned int JpegHuffmanDecoder::ReadTable (JpegDecoder &decoder)
{
  // We declare this here because MSVC++ does not handle for
  // statement scoping rules correctly.
  unsigned int jj ;

  // B.2.4.2
  UBYTE1 huffbits [JpegMaxHuffmanCodeLength] ;

  unsigned int count = 0 ; // Count of codes in the Huffman table.

  // Read the 16 1-byte length counts and count the number of
  // codes in the table.
  for (jj = 0 ; jj < JpegMaxHuffmanCodeLength ; ++ jj)
  {
    // These values are called Li in the standard.
    huffbits [jj] = decoder.ReadByte () ;
    count += huffbits [jj] ;
  }
  if (count > JpegMaxNumberOfHuffmanCodes)
    throw EJpegBadData ("Huffman count > 256") ;

  // Read the Huffman values.
  for (jj = 0 ; jj < count ; ++ jj)
  {
    // These values are called Vij in the standard.
    huff_values [jj] = decoder.ReadByte () ;
  }

  // Generate the structures for Huffman decoding.
  MakeTable (huffbits) ;

  table_defined = true ; // This table can now be used.

  return JpegMaxHuffmanCodeLength + count ;
}

//
//  Description:
//


//    This function generates the data used for Huffman decoding.
//
//    The implicit outputs are the member variables mincode [n],
//    maxcode [n] and valptr [n]. These are the minimum Huffman Code of
//    length n+1, the maximum Huffman Code of length n+1, and the index
//    into huff_values [] for the first value with a Huffman code of length
//    n+1.
//
//  Parameters:
//    huffbits: The count of Huffman codes of length n+1)
//
void JpegHuffmanDecoder::MakeTable (UBYTE1 huffbits [JpegMaxHuffmanCodeLength])
{
  // We have to declare the loop indices here because MSVC++ does not
  // handle scoping in for statements correctly.
  unsigned int ii, jj, kk ;

  // These values in these arrays correspond to the elements of the
  // "values" array. The Huffman code for values [N] is huffcodes [N]
  // and the length of the code is huffsizes [N].

  UBYTE2 huffcodes [JpegMaxNumberOfHuffmanCodes] ;
  unsigned int huffsizes [JpegMaxNumberOfHuffmanCodes + 1] ;

  // Section C.2 Figure C.1
  // Convert the array "huff_size" containing the count of codes
  // for each length into the array "huffsizes" containing the length for each
  // size.
  for (ii = 0, kk = 0 ; ii < JpegMaxHuffmanCodeLength ; ++ ii)
  {
    for (int jj = 0 ; jj < huffbits [ii] ; ++ jj)
    {
      huffsizes [kk] = ii + 1 ;
      ++ kk ;
    }
    huffsizes [kk] = 0 ;
  }
```

closed To And Understood By Me

Name	Date	Signature:
Name	Date	Date:

Subject:

```
// Section C.2 Figure C.2
// Calculate the Huffman code for each Huffman value.
UBYTE2 code = 0 ;
unsigned int si ;
for (kk = 0, si = huffsizes [0] ;
     huffsizes [kk] != 0 ;
     ++ si, code <<= 1)
{
   for ( ; huffsizes [kk] == si ; ++ code, ++ kk)
   {
      huffcodes [kk] = code ;
   }
}

// Section F.2.2. Figure F.15
// Create three arrays.
// mincode [n] : The smallest Huffman code of length n + 1.
// maxcode [n] : The largest Huffman code of length n + 1.
// valptr [n] : Index into the values array. First value with a code
//              of length n + 1.
for (ii=0, jj=0 ; ii < JpegMaxHuffmanCodeLength ; ++ ii)   for 1 ~ 16
{
   // ii is the index into Huffman code lengths
   // jj is the index into Huffman code values
   if (huffbits [ii] != 0)
   {
      // The jj'th Huffman value is the first with a Huffman code
      // of length ii.
      valptr [ii] = jj ;
      mincode [ii] = huffcodes [jj] ;
      jj += huffbits [ii] ;
      maxcode [ii] = huffcodes [jj - 1] ;
   }
   else
   {
      // There are no Huffman codes of length (ii + 1).
      maxcode [ii] = -1 ;
      // An illegal value > maxcode[]
      mincode [ii] = JpegMaxNumberOfHuffmanCodes + 1 ;
      valptr [ii] = 0 ;
   }
}
return ;
}

//
//  Description:
//
//    This function decodes the next Huffman-encoded value in the input
//    stream.
//
//  Parameters:
//    decoder:  The JPEG decoder that owns the Huffman table.

//
int JpegHuffmanDecoder::Decode (JpegDecoder &decoder)
{
   // This function decodes the next byte in the input stream using this
   // Huffman table.

   // Section A F.2.2.3 Figure F.16

   UBYTE2 code = decoder.NextBit () ;
   int codelength ; // Called I in the standard.

   // Here we are taking advantage of the fact that 1 bits are used as
   // a prefix to the longer codes.
   for (codelength = 0 ;
        (code > maxcode [codelength] && codelength <
         JpegMaxHuffmanCodeLength) ;
        ++ codelength)
   {
      code = (UBYTE2) ((code << 1) | decoder.NextBit ()) ;
   }

   if (codelength >= JpegMaxHuffmanCodeLength)
    throw EJpegBadData ("Bad Huffman Code Length") ;

   // Now we have a Huffman code of length (codelength + 1) that
   // is somewhere in the range
   // mincode [codelength]..maxcode [codelength].

   // This code is the (offset + 1)'th code of (codelength + 1) ;
   int offset = code - mincode [codelength] ;

   // valptr [codelength] is the first code of length (codelength + 1)
   // so now we can look up the value for the Huffman code in the table.
   int index = valptr [codelength] + offset ;
   return huff_values [index] ;
}
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2010/8/18

17150

Subject: Gamma Correction Source code study

```
--------------------------------------------------------------------------------
--
- gama_corret  : gamma correction
-
- designer : Albert Wang
-
- description  : a lookup table is used for gamma correction that includes
-       1. gamma correct lookup table initialization via I2C
-       2. panel data is processed via lookup table for gamma correct
-       3. bypass gamma correction
-       Group delay = 2 pclk for gamma correction
-
- Date      : Aug 02 1998   first creation
-
- Version   : 2.0
-
- History   : Sep 23 1998
-       replace tsmc sram with artisan sram
-       Oct. 28 1999
-       replace 256x8 with 256x10
--------------------------------------------------------------------------------
-
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

- pragma translate_off
library scaling_lib;
library tsmc_macro_025_lib ;
use   scaling_lib.scaling_components_pkg.all;
- pragma translate_on

- escalade ports -begin
entity gc_top is
  port (rstn :    in    std_logic;
        pclk :    in    std_logic;
        pvs : in    std_logic;
        IIC_DIRW :    in    std_logic;
        GC_use :  in    std_logic;
        rgb_in :  in    std_logic_vector(23 downto 0);
        host_ibus :   in    std_logic_vector(17 downto 0);
        softzero :    in    std_logic;
        tis : in    std_logic;
        tms : in    std_logic;
        TC :  in    std_logic;
        TD_GC_F : in    std_logic;
        TD_GC :   in    std_logic_vector(1 downto 0);
        TQ_GC :   out   std_logic_vector(1 downto 0);
        TQ_GC_F : out   std_logic;
        TD_LUT :  in    std_logic_vector(2 downto 0);
        TQ_LUT :  out   std_logic_vector(2 downto 0);
        CLUT_ALPHA :  out   std_logic_vector(3 downto 0);
```

sclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 11/5/15

Subject: _____ 17131

```
        RCAB_CBUS :   out   std_logic_vector(15 downto 0);
        GCAB_CBUS :   out   std_logic_vector(15 downto 0);
        BCAB_CBUS :   out   std_logic_vector(15 downto 0);
        DEDIM_LIGHT : out   std_logic_vector(3 downto 0);
        DM_V_ASTART : out   std_logic_vector(10 downto 0);
        DM_V_AEND :   out   std_logic_vector(10 downto 0);
        DM_H_ASTART : out   std_logic_vector(10 downto 0);
        DM_H_AEND :   out   std_logic_vector(10 downto 0);
        LUT_CBUS :    out   std_logic_vector(8 downto 0);
        rgb_out : out   std_logic_vector(29 downto 0));

ARCHITECTURE BEHAVIORAL OF gc_top IS

COMPONENT gamma_correct
  PORT (
        rstn        : in  std_logic;    -- reset signal
        PClk        : in  std_logic;    -- panel clock
        GC_tbl_write: in  std_logic;    -- I2C GC_tbl write pulse from host_if
        GC_value    : in  std_logic_vector(7  downto 0); -- GC initial value fr
om host_if
        GC_use      : in  std_logic;    -- use gamma_correction from hostif
        Data_in     : in  std_logic_vector(7  downto 0); -- panel input data fr
om scaling
        GC_tbl_out  : in  std_logic_vector(9  downto 0); -- GC table data outpu
t
        GC_tbl_in   : out std_logic_vector(9  downto 0); -- input data for GC t
able
        GC_tbl_addr : out std_logic_vector(7  downto 0); -- GC_tbl address
        GC_tbl_ceb  : out std_logic;    -- ramceb for GC table
        GC_tbl_web  : out std_logic;    -- ramweb for GC table
        GC_tbl_clk  : out std_logic;      -- ramclk for GC table
        Data_out    : out std_logic_vector(9  downto 0)  -- panel data out
      );
END COMPONENT;

COMPONENT sram_256x10
  PORT (
        in1         : in  std_logic_vector(9 downto 0); -- data input
        clk         : in  std_logic;    -- clock input
        cen         : in  std_logic;    -- chip enable
        oen         : in  std_logic;    -- output enable
        wen         : in  std_logic;    -- write enable
        a           : in  std_logic_vector(7  downto 0); -- address
        tis         : in  std_logic;    -- test input select; from tcb test_act
ve
        tms         : in  std_logic;    -- test mode select; from tc
        tqoen       : in  std_logic;    -- test data output drive enable; from
tcb
        serial_in   : in  std_logic;    -- test data input
        out1        : out std_logic_vector(9 downto 0); -- data output
        serial_out  : out std_logic     -- test data output
    );
END COMPONENT;
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2003/5/15

17132

Subject: _____

```
OMPONENT delay comp
 PORT (
       pclk          : in  std_logic;     -- panel clock
       phs           : in  std_logic;     -- panel H_sync
       ph_active     : in  std_logic;     -- panel horizontal active
       pde           : in  std_logic;     -- panel horizontal display enable
       pvs           : in  std_logic;     -- panel V_sync
       pv_active     : in  std_logic;     -- panel vertical active
       pvde          : in  std_logic;     -- panel vertical display enable
       phs_out       : out std_logic;     -- panel H_sync output
       ph_active_out: out std_logic;      -- panel horizontal active output
       pde_out       : out std_logic;     -- panel horizontal display enable outp
t
       pvs_out       : out std_logic;     -- panel V_sync output
       pv_active_out: out std_logic;      -- panel vertical active output
       pvde_out      : out std_logic      -- panel vertical display enable output
       );
PORT (
       CLK           : in  std_logic;
       RSTN          : in  std_logic;
       SZERO         : in  std_logic;
       IIC_DIRW      : in  std_logic;
       PVS           : in  std_logic;
       RCAB_CBUS : out   std_logic_vector(15 downto 0);
       GCAB_CBUS : out   std_logic_vector(15 downto 0);
       BCAB_CBUS : out   std_logic_vector(15 downto 0);
       DEDIM_LIGHT : out std_logic_vector(3  downto 0);
       CLUT_ALPHA  : out std_logic_vector(3  downto 0);
       DM_V_ASTART : out std_logic_vector(10 downto 0);
       DM_V_AEND   : out std_logic_vector(10 downto 0);
       DM_H_ASTART : out std_logic_vector(10 downto 0);
       DM_H_AEND   : out std_logic_vector(10 downto 0);
       LUT_CBUS    : out std_logic_vector(8  downto 0);
       GC_CBUS     : out std_logic_vector(10 downto 0);
       HOST_IBUS : in    std_logic_vector(17 downto 0));
IPONENT;
-------------------------------------------------------------------------------
--
 host interface
-------------------------------------------------------------------------------
--
gnal gc_cbus          : std_logic_vector(10 downto  0);
gnal data_value       : std_logic_vector(7  downto  0);
gnal grwaddr_write        : std_logic;
gnal ggwaddr_write        : std_logic;
gnal gbwaddr_write        : std_logic;
-------------------------------------------------------------------------------
-
 gamma correction for r input
-------------------------------------------------------------------------------
-
```

)isclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date:

Subject: _____

```
signal r_gc_out          : std_logic_vector(9  downto  0);
signal r_gc_in           : std_logic_vector(9  downto  0);
signal r_gc_addr         : std_logic_vector(7  downto  0);
signal r_gc_ceb          : std_logic;
signal r_gc_web          : std_logic;
signal r_gc_clk          : std_logic;
-------------------------------------------------------------------------------
---
-- gamma correction for g input
-------------------------------------------------------------------------------
---
signal g_gc_out          : std_logic_vector(9  downto  0);
signal g_gc_in           : std_logic_vector(9  downto  0);
signal g_gc_addr         : std_logic_vector(7  downto  0);
signal g_gc_ceb          : std_logic;
signal g_gc_web          : std_logic;
signal g_gc_clk          : std_logic;
-------------------------------------------------------------------------------
---
-- gamma correction for b input
-------------------------------------------------------------------------------
---
signal b_gc_out          : std_logic_vector(9  downto  0);
signal b_gc_in           : std_logic_vector(9  downto  0);
signal b_gc_addr         : std_logic_vector(7  downto  0);
signal b_gc_ceb          : std_logic;
signal b_gc_web          : std_logic;

-------------------------------------------------------------------------------
---
--
-- Host interface part; decode I2C address and data
--
-------------------------------------------------------------------------------
---
i_host_gc: hostif_gc
  GENERIC MAP
        (
       ADDR_SIZE => 8,
       DATA_SIZE => 8
      )
  PORT MAP
       (
        CLK          => pclk,
        RSTN         => rstn,
        SZERO        => softzero,
        IIC_DIRW     => IIC_DIRW,
        PVS          => pvs ,
        RCAB_CBUS    => RCAB_CBUS,
        GCAB_CBUS    => GCAB_CBUS,
        BCAB_CBUS    => BCAB_CBUS,
        DEDIM_LIGHT  => DEDIM_LIGHT,
        CLUT_ALPHA   => CLUT_ALPHA,
```

Disclosed To And Understood By Me

_____ _____
Name Date

_____ _____
Name Date

Signature: blair

Date: 2011/5/15

```
      DM_V_ASTART => DM_V_ASTART,
      DM_V_AEND   => DM_V_AEND,
      DM_H_ASTART => DM_H_ASTART,
      DM_H_AEND   => DM_H_AEND,
      LUT_CBUS    => LUT_CBUS,
      GC_CBUS     => gc_cbus,
      HOST_IBUS   => host_ibus
     );
ata_value <= gc_cbus(7 downto 0);
rwaddr_write <=gc_cbus(8);
gwaddr_write <=gc_cbus(9);
bwaddr_write <=gc_cbus(10);
-------------------------------------------------------------------------------
--
-
- gamma correction processing for r input
-
-------------------------------------------------------------------------------
-
-- gc: gamma_correct
PORT MAP
     (
      rstn         => rstn,
      PClk         => pclk,
      GC_tbl_write=> grwaddr_write,
      GC_value     => data_value,
      GC_use       => GC_use,
      Data_in      => rgb_in(23  downto 16),
      GC_tbl_out   => r_gc_out,
      GC_tbl_in    => r_gc_in,
      GC_tbl_addr  => r_gc_addr,
      GC_tbl_ceb   => r_gc_ceb,
      GC_tbl_web   => r_gc_web,
      GC_tbl_clk   => r_gc_clk,
      Data_out     => rgb_out(29  downto 20)
     );
--TDMS_WRITE
      inl          => r_gc_in,
      clk          => r_gc_clk,
      cen          => r_gc_ceb,
      oen          => softzero,
      wen          => r_gc_web,
      a            => r_gc_addr,
      tis          => tis,
      tms          => tms,
      tqoen        => softzero,
      serial_in    => TD_LUT(0),
      out1         => r_gc_out,
      serial_out   => TQ_LUT(0)
     );
```

(RGB pulse DATA_IN)

(GBWADDR_write GGWADDR_WRITE, GRWADDR_WRITE, DATA_IN)

DATA-IN

GRWADDR_~~WRITE~~EN AND WR_PUL

---

sclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 2/5/15

Subject: _____

```
--
--
-- gamma correction processing for g input
--
-------------------------------------------------------------------------------
---
_g_gc: gamma_correct
 PORT MAP
       (
        rstn         => rstn,
        PClk         => pclk,
        GC_tbl_write=> ggwaddr_write,
        GC_value     => data_value,
        GC_use       => GC_use,
        Data_in      => rgb_in(15 downto 8),
        GC_tbl_out   => g_gc_out,
        GC_tbl_in    => g_gc_in,
        GC_tbl_addr  => g_gc_addr,
        GC_tbl_ceb   => g_gc_ceb,
        GC_tbl_web   => g_gc_web,
        GC_tbl_clk   => g_gc_clk,
        Data_out     => rgb_out(19 downto 10)
       );

_g_gc_tbl: sram_256x10
 PORT MAP
       (
        in1          => g_gc_in,
        clk          => g_gc_clk,
        cen          => g_gc_ceb,
        oen          => softzero,
        wen          => g_gc_web,
        a            => g_gc_addr,
        tis          => tis,
        tms          => tms,
        tqoen        => softzero,
        serial_in    => TD_LUT(1),
        out1         => g_gc_out,
        serial_out   => TQ_LUT(1)
       );
-------------------------------------------------------------------------------
---
--
-- gamma correction processing for b input
--
-------------------------------------------------------------------------------
--

        rstn         => rstn,
        PClk         => pclk,
        GC_tbl_write=> gbwaddr_write,
        GC_value     => data_value,
        GC_use       => GC_use,
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 20/5/15

Subject: _____

```
      Data_in     => rgb_in(7 downto 0),
      GC_tbl_out  => b_gc_out,
      GC_tbl_in   => b_gc_in,
      GC_tbl_addr => b_gc_addr,
      GC_tbl_ceb  => b_gc_ceb,
      GC_tbl_web  => b_gc_web,
      GC_tbl_clk  => b_gc_clk,
      Data_out    => rgb_out(9 downto 0)
     );

b_gc_tbl: sram_256x10
PORT MAP
     (
      inl         => b_gc_in,
      clk         => b_gc_clk,
      cen         => b_gc_ceb,
      oen         => softzero,
      wen         => b_gc_web,
      a           => b_gc_addr,
      tis         => tis,
      tms         => tms,
      tqoen       => softzero,
      serial_in   => TD_LUT(2),
      out1        => b_gc_out,
      serial_out  => TQ_LUT(2)
     );

-------------------------------------------------------------------------------
--
-
- group delay compensation
-
-------------------------------------------------------------------------------
--
-i_delay_comp: delay_comp
-  PORT MAP
-       (
-  pclk        => pclk,
-  phs     => phs,
-  ph_active   => ph_active,
-  pde     => pde,
-  pvs     => pvs,
-  pv_active   => pv_active,
-  pvde        => pvde,
-  phs_out     => phs_out,
-  ph_active_out   => ph_active_out,
-  pde_out     => pde_out,
-  pvs_out     => pvs_out,
-  pv_active_out   => pv_active_out,
-  pvde_out    => pvde_out
-       );

'D BEHAVIORAL;
```

)isclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 12/12/02

Subject:

```
File: D:\projects\ttfcII\escalade_db\proj_zuracII\@CODEGEN\gamma_correct_2.vhd
2000/3/9, 05:16:14PM
-- Copyright (c) 1994-1999 Escalade Corporation. All rights reserved.
-- Block proj_zuracII/gamma_correct:BEHAVIORAL
--   7 inputs
--   6 outputs
--   0 inouts
--
-- RTL code generated by DesignBook LAN
--
-- Title:            proj_zuracII/gamma_correct:BEHAVIORAL
-- Author:           albert
-- Date:             12/21/99 19:43
-- Description:
--
--
--
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
library ttfc_pkgs_lib;
use ttfc_pkgs_lib.constant_pkg.all ;
use ttfc_pkgs_lib.reg_table_pkg.all ;

-- pragma translate_off
--library MATRIX_Lib ;
library IIC_Lib ;
library FBC_Lib ;
library tsmc_macro_025_lib;
library auto_check_lib;
library tsmc_io_025_lib;
-- library ttfc_gate_lib ;
-- use ttfc_gate_lib.all ;
-- pragma translate_on

-- escalade endheader

--------------------------------------------------------------------------------
--
-- gama_corret   : gamma correction
--
-- designer      : Albert Wang
--
-- description   : a lookup table is used for gamma_correction that includes
--                 1. gamma_correct lookup table initialization via I2C
--                 2. panel data is processed via lookup table for gamma_correct
--                 3. bypass gamma_correction
--                 Group delay = 3 pclk for gamma_correction
--
-- Date          : Aug 02 1998    first creation
--
-- Version       : 2.0
--
-- History       : extend gamma table from 8 bit to 10 bit, need two write cycle
--                 to collect 10 bit data from I2C
--------------------------------------------------------------------------------
--
library ieee;
use     ieee.std_logic_1164.all;
use     ieee.std_logic_unsigned.all;
-- escalade ports -begin
entity gamma_correct is
    port (rstn :          in     std_logic;
          PClk :          in     std_logic;
          GC_tbl_write :         in     std_logic;
          GC_value :      in     std_logic_vector(7 downto 0);
          GC_use :        in     std_logic;
          Data_in :       in     std_logic_vector(7 downto 0);
          GC_tbl_out :    in     std_logic_vector(9 downto 0);
          GC_tbl_in :     out    std_logic_vector(9 downto 0);
          GC_tbl_addr :   out    std_logic_vector(7 downto 0);
          GC_tbl_ceb :    out    std_logic;
          GC_tbl_web :    out    std_logic;
          GC_tbl_clk :    out    std_logic;
          Data_out :      out    std_logic_vector(9 downto 0));
end gamma_correct;
-- escalade ports -end

ARCHITECTURE BEHAVIORAL OF gamma_correct IS

component CounTURES
  port
  (
    Clk           : in     std_logic;
    Reset         : in     std_logic;
    Enable        : in     std_logic;
    Count         : out    std_logic_vector(8 downto 0)
  );
end component;

SIGNAL data_in_D1       : std_logic_vector(7 downto 0);
SIGNAL data_in_D2       : std_logic_vector(7 downto 0);
```

isclosed To And Understood By Me

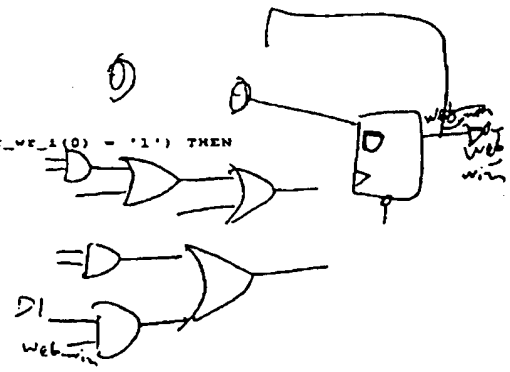
Name	Date	Signature: blair
Name	Date	Date: 2022/5/18

17138

Subject: _____

File: D:\projects\ttfcII\escalade_db\proj_zuracII\@CODEGEN\gamma_correct_2.vhd
2000/3/9, 05:16:14PM

```
SIGNAL cnt_rst           : std_logic;
SIGNAL GC_tbl_write_D1   : std_logic;
SIGNAL GC_tbl_write_D2   : std_logic;
SIGNAL GC_tbl_addr_wr_i  : std_logic_vector(8 downto 0);
SIGNAL GC_tbl_addr_wr    : std_logic_vector(7 downto 0);
SIGNAL GC_tbl_addr_rd    : std_logic_vector(7 downto 0);
SIGNAL lreg              : std_logic_vector(7 downto 0);
SIGNAL GC_tbl_ceb_wr     : std_logic;
SIGNAL GC_tbl_ceb_rd     : std_logic;
SIGNAL GC_tbl_web_wr     : std_logic;
SIGNAL GC_tbl_web_rd     : std_logic;
SIGNAL mode_sel          : std_logic;
SIGNAL web_win           : std_logic;

BEGIN

-------------------------------------------------------------------------------
--
-- Gamma_correct table initialization
--
-------------------------------------------------------------------------------
--
cnt_rst <= NOT rstn;

-- gamma_correct table writing address generation
-- after GC_value is written into lookup table
-- address is increased by delay of GC_tbl_write
i_addr_cnt: CountURES
  PORT MAP
       (
        Clk      => PClk,
        Reset    => cnt_rst,
        Enable   => GC_tbl_write_D2,
        Count    => GC_tbl_addr_wr_i
       );

-- actual GC table address increase by two write pulse
GC_tbl_addr_wr <= GC_tbl_addr_wr_i(8 downto 1);

-- latch low nibble when GC_tbl_addr_wr_i(0) is low
PROCESS(PClk,
  BEGIN
       IF (PClk'event AND PClk = '1') THEN
         IF (rstn = '0') THEN
           lreg <= (others => '0');
         ELSE
           IF (GC_tbl_addr_wr_i(0) = '0' AND GC_tbl_write = '1') THEN
             lreg <= GC_value;
           END IF;
         END IF;
       END IF;
  END PROCESS;

-- ramweb for write
PROCESS(PClk)
  BEGIN
       IF (PClk'event AND PClk = '1') THEN
         IF (rstn = '0') THEN
           web_win <= '1';
         ELSIF (GC_tbl_write = '1' AND GC_tbl_addr_wr_i(0) = '1') THEN
           web_win <= '0';
         ELSIF (GC_tbl_write_D1 = '0') THEN
           web_win <= '1';
         ELSE
           web_win <= web_win;
         END IF;
       END IF;
  END PROCESS;

GC_tbl_web_wr <= web_win;

PROCESS(PClk)
  BEGIN
       IF (PClk'event AND PClk = '1') THEN
         GC_tbl_write_D1 <= GC_tbl_write;
         GC_tbl_write_D2 <= GC_tbl_write_D1;
         IF (GC_tbl_addr_wr_i = '1') THEN
           GC_tbl_in <= GC_value(3 DOWNTO 0) & lreg;
         END IF;
       END IF;
  END PROCESS;

-- ramceb for write
PROCESS(PClk)
  BEGIN
       IF (PClk'event AND PClk = '0') THEN
         IF (GC_tbl_addr_wr_i(0) = '1') THEN
           GC_tbl_ceb_wr <= (NOT GC_tbl_write_D1);
         ELSE
           GC_tbl_ceb_wr <= '1';
         END IF;
```



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/3/15

File: D:\projects\ttfcII\escalade_db\proj_zuracII\@CODEGEN\gamma_correct_2.vhd
2000/3/9, 05:16:14PM

```
      END IF;
  END PROCESS;

-------------------------------------------------------------------------------
--
-- Gamma_correct manipulation part
--
-------------------------------------------------------------------------------
--
-- delay part
PROCESS(PClk)
  BEGIN
      IF (PClk'event AND PClk = '1') THEN
        data_in_D1 <= data_in;
        data_in_D2 <= data_in_D1;
      END IF;
  END PROCESS;

-- map graphic input data into GC_tbl address
GC_tbl_addr_rd <= data_in_D1;

-- according to GC_use assign GC ramceb for read
PROCESS(GC_use, PClk)
  BEGIN
      IF (GC_use = '1') THEN
        GC_tbl_ceb_rd <= PClk;
        GC_tbl_web_rd <= '1';
      ELSE
        GC_tbl_ceb_rd <= '1';
        GC_tbl_web_rd <= '1';
      END IF;
  END PROCESS;

-------------------------------------------------------------------------------
--
-- Gamma_correct output selection
--
-------------------------------------------------------------------------------
--
-- generate ramceb, ramweb and ram address for read/write

mode_sel <= NOT web_win;

PROCESS(mode_sel, GC_tbl_addr_rd, GC_tbl_addr_wr)
  BEGIN
      IF (mode_sel = '1') THEN
        GC_tbl_addr <= GC_tbl_addr_wr;
      ELSE
        GC_tbl_addr <= GC_tbl_addr_rd;
      END IF;
  END PROCESS;

PROCESS(mode_sel, GC_tbl_ceb_rd, GC_tbl_ceb_wr)
  BEGIN
      IF (mode_sel = '1') THEN
        GC_tbl_ceb <= GC_tbl_ceb_wr;
      ELSE
        GC_tbl_ceb <= GC_tbl_ceb_rd;
      END IF;
  END PROCESS;

PROCESS(mode_sel, GC_tbl_web_rd, GC_tbl_web_wr)
  BEGIN
      IF (mode_sel = '1') THEN
        GC_tbl_web <= GC_tbl_web_wr;
      ELSE
        GC_tbl_web <= GC_tbl_web_rd;
      END IF;
  END PROCESS;

-- assign data_out that is from GC_tbl or direct input
PROCESS(PClk)
  BEGIN
      IF (PClk'event AND PClk = '1') THEN
        IF (GC_use = '1') THEN
          Data_out <= GC_tbl_out;
        ELSE
          Data_out <= data_in_D2 & '0' & '0';
        END IF;
      END IF;
  END PROCESS;

GC_tbl_clk <= PClk;

END BEHAVIORAL;
```

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2000/5/15

17:10

Subject: _____

File: D:\projects\ttfcII\escalade_db\proj_turacII\@CODEGEN\CountURE9.vhd 2000/1/31, 08:28:06PM

```
--------------------------------------------------------------------------------
--
-- designer      : Albert Wang
--
-- description   : 9 bits Up counter, Synchronous Reset, Enable input
--
-- history       : Aug 03 1998   first creation
--
--------------------------------------------------------------------------------
--
-- escalade ports -begin
entity CountURE9 is
    port (Clk    : in     std_logic;
          Reset  :        in     std_logic;
          Enable :        in     std_logic;
          Count  :        out    std_logic_vector(8 downto 0));
end CountURE9;
-- escalade ports -end

architecture ver1 of CountURE9 is

  signal c       : std_logic_vector(8 downto 0);
  signal cyLo    : std_logic;

begin

  clo_process : process(Clk)
  begin
    if (Clk'event AND CLK = '1') then
      if Reset = '1' then
        c(3 downto 0) <= "0000";
      else
      if Enable = '1' then
        c(0) <= not c(0);
        if c(0) = '1' then
          c(1) <= not c(1);
          if c(1) = '1' then
            c(2) <= not c(2);
            if c(2) = '1' then
              c(3) <= not c(3);
            end if;
          end if;
        end if;
      end if;
      end if;
    end if;
  end process;

  cylo_process : process(Clk)
  begin
    if (Clk'event AND CLK = '1') then
      if Reset = '1' then
        cyLo <= '0';
      else
        case c(3 downto 0) is
          when "1111" => cyLo <= not Enable;
          when "1110" => cyLo <= Enable;
          when others  => cyLo <= '0';
        end case;
      end if;
    end if;
  end process;

  chi_process : process(Clk)
  begin
    if (Clk'event AND CLK = '1') then
      if Reset = '1' then
        c(8 downto 4) <= "00000";
      else
      if ((Enable = '1') and (cyLo = '1')) then
        c(4) <= not c(4);
        if c(4) = '1' then
          c(5) <= not c(5);
          if c(5) = '1' then
            c(6) <= not c(6);
            if c(6) = '1' then
              c(7) <= not c(7);
              if c(7) = '1' then
                c(8) <= not c(8);
              end if;
            end if;
          end if;
        end if;
      end if;
      end if;
    end if;
  end process;

  Count <= c;

end ver1;
```

Disclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date:

Subject: hostif_gc.vhdl study

File: D:\projects\ttfc!I\escalade_db\proj_zurac!!\zSIM_TOP_V1\blk\sim\hostif_gc.vhd 2000/3/9, 05:16:14PM

```
-- Author:          gshwang
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
library ttfc_pkgs_lib;
use ttfc_pkgs_lib.constant_pkg.all ;
use ttfc_pkgs_lib.reg_table_pkg.all ;

library IIC_Lib ;
library FBC_Lib ;
library tsmc_macro_025_lib;
library auto_check_lib;
library tsmc_io_025_lib;

library IEEE,ttfc_pkgs_lib;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use ttfc_pkgs_lib.constant_pkg.all ;
use ttfc_pkgs_lib.reg_table_pkg.all ;-- escalade ports -begin
entity hostif_gc is
    generic (ADDR_SIZE :        integer := 8;
        DATA_SIZE :     integer := 8);
    port (CLK : in    std_logic;
          RSTN :        in    std_logic;
          SZERO :       in    std_logic;
          IIC_DIRW :    in    std_logic;
          PVS : in    std_logic;
          RCAB_CBUS :   out   std_logic_vector(15 downto 0);
          GCAB_CBUS :   out   std_logic_vector(15 downto 0);
          BCAB_CBUS :   out   std_logic_vector(15 downto 0);
          DEDIM_LIGHT : out   std_logic_vector(3 downto 0);
          CLUT_ALPHA :  out   std_logic_vector(3 downto 0);
          DM_V_ASTART : out   std_logic_vector(10 downto 0);
          DM_V_AEND :   out   std_logic_vector(10 downto 0);
          DM_H_ASTART : out   std_logic_vector(10 downto 0);
          DM_H_AEND :   out   std_logic_vector(10 downto 0);
          LUT_CBUS :    out   std_logic_vector(8 downto 0);
          GC_CBUS :     out   std_logic_vector(10 downto 0);
          HOST_IBUS :   in    std_logic_vector(17 downto 0));
end hostif_gc;
-- escalade ports -end


architecture behavioral of hostif_gc  is
-------------------------
-- component declare  --
-------------------------
-- for 8 bits or less register
component DEC_REG_1ADDR
    generic (
                ADDR_SIZE       : integer := 8; -- address bits size
                DATA_SIZE       : integer := 8; -- data transfer bits size
        REGDATA_SIZE    : integer := 8); -- reg used bits size
    port (
                CLK             : in  std_logic;
                RSTN            : in  std_logic;
                SZERO           : in  std_logic;
        IIC_DIRW :      in    std_logic;
                VS_PUL          : in  std_logic;
                HOST_IBUSN      : in  std_logic_vector(DATA_SIZE-1       downto
0);
                REG_EN          : in  std_logic;
                REG_INI         : in  std_logic_vector(REGDATA_SIZE-1    downto
0);
                REG_VALUE       : out std_logic_vector(REGDATA_SIZE-1    downto
0);
                REG_OUT         : out std_logic_vector(DATA_SIZE-1       downto
0)
        );
end component ;

-- for 11 bits register
component DEC_REG_2ADDRUP
    port (
                CLK             : in  std_logic;
                RSTN            : in  std_logic;
                SZERO           : in  std_logic;
                IIC_DIRW    : in  std_logic;
                VS_PUL          : in  std_logic;
                HOST_IBUSN      : in  std_logic_vector(9  downto 0);
                REGH_EN         : in  std_logic;
                REGL_EN         : in  std_logic;
                REG_INI         : in  std_logic_vector(10 downto 0);
                REG_VALUE       : out std_logic_vector(10 downto 0);
                REG_OUT         : out std_logic_vector(7  downto 0));
end component;

-- CLUT_ALPHA register value        01/11/99
signal  CLUT_ALPHA_DATA : std_logic_vector(7 downto 0); -- reg 86
```

Vertical start pulse

← [RGB_pulse & DATA]

RD & WR & DATA_IN & ADDR_IN

Disclosed To And Understood By Me

Name Date Signature: Blair

Name Date Date: 2000/5/17

17142.
ubject:

File: D:\projects\ttfcII\escalade_db\proj_zuracII\:SIM_TOP_V1\blk\sim3\hostif_gc
.vnd 2000/3/9, 05:16:14PM

```
-- DE_DIM register data                26/10/99
signal   DMH_START_DATA    : std_logic_vector(10 downto 0); -- reg 49,48
signal   DMH_END_DATA      : std_logic_vector(10 downto 0); -- reg 4B,4A
signal   DMV_START_DATA    : std_logic_vector(10 downto 0); -- reg 4D,4C
signal   DMV_END_DATA      : std_logic_vector(10 downto 0); -- reg 4F,4E
signal   DE_LIGHT_DATA     : std_logic_vector(3 downto 0); -- reg 4F MSB 4-bit
signal   DMV_ENDH_DATA     : std_logic_vector(7 downto 0); -- reg 4F
signal   DMV_ENDL_DATA     : std_logic_vector(7 downto 0); -- reg 4E
-- brightness & contrast register
signal   RCONTRAST_DATA    : std_logic_vector( 7 downto 0); -- reg AE, ZuracII
signal   RBRIGHTNESS_DATA          : std_logic_vector( 7 downto 0); -- reg AF
signal   GCONTRAST_DATA    : std_logic_vector( 7 downto 0); -- reg B0, ZuracII
signal   GBRIGHTNESS_DATA          : std_logic_vector( 7 downto 0); -- reg B1
signal   BCONTRAST_DATA    : std_logic_vector( 7 downto 0); -- reg B7, ZuracII
signal   BBRIGHTNESS_DATA          : std_logic_vector( 7 downto 0); -- reg B8
-- CLUT_ALPHA register value          01/11/99
signal   CLUT_ALPHA_OUT    : std_logic_vector(7  downto 0); -- reg B6
-- DE_DIM register output data 26/10/99
signal   DMH_START_OUT     : std_logic_vector(7  downto 0); --
signal   DMH_END_OUT               : std_logic_vector(7  downto 0); --
signal   DMV_START_OUT     : std_logic_vector(7  downto 0); --
signal   DMV_END_OUT               : std_logic_vector(7  downto 0); --
-- brightness & contrast register output data
signal   RCONTRAST_OUT     : std_logic_vector(7  downto 0);
signal   RBRIGHTNESS_OUT   : std_logic_vector(7  downto 0);
signal   GCONTRAST_OUT     : std_logic_vector(7  downto 0);
signal   GBRIGHTNESS_OUT   : std_logic_vector(7  downto 0);
signal   BCONTRAST_OUT     : std_logic_vector(7  downto 0);
signal   BBRIGHTNESS_OUT   : std_logic_vector(7  downto 0);
-- CLUT_ALPHA register value          01/11/99
signal CLUT_ALPHA_EN       : std_logic;
-- DE_DIM register data               26/10/99
signal DMH_STARTL_EN       : std_logic;
signal DMH_STARTH_EN       : std_logic;
signal DMH_ENDL_EN         : std_logic;
signal DMH_ENDH_EN         : std_logic;
signal DMV_STARTL_EN       : std_logic;
signal DMV_STARTH_EN       : std_logic;
signal DMV_ENDL_EN         : std_logic;
signal DMV_ENDH_EN         : std_logic;
-- brightness & contrast register
signal   RCONTRAST_EN      : std_logic;
signal   RBRIGHTNESS_EN    : std_logic;
signal   GCONTRAST_EN      : std_logic;
signal   GBRIGHTNESS_EN    : std_logic;
signal   BCONTRAST_EN      : std_logic;
signal   BBRIGHTNESS_EN    : std_logic;
signal   GRWADDR_EN        : std_logic;
signal   GGWADDR_EN        : std_logic;
signal   GBWADDR_EN        : std_logic;
signal   LUTWADDR_EN       : std_logic;
signal   GRWADDR_write     : std_logic;
signal   GGWADDR_write     : std_logic;
signal   GBWADDR_write     : std_logic;
signal   LUTWADDR_write    : std_logic;
----------------------------
-- initial value table   --
----------------------------
-- CLUT_ALPHA register value          01/11/99
signal   CLUT_ALPHAINI     : std_logic_vector(7 downto 0);
-- DE_DIM register initial value              26/10/99
signal   DMH_STARTINI      : std_logic_vector(10 downto 0);
signal   DMH_ENDINI        : std_logic_vector(10 downto 0);
signal   DMV_STARTINI      : std_logic_vector(10 downto 0);
signal   DMV_ENDHINI       : std_logic_vector(7 downto 0);
signal   DMV_ENDLINI       : std_logic_vector(7 downto 0);
-- ZuracII, initial value             26/10/99
signal   RCONTRASTINI              : std_logic_vector(7  downto 0);
signal   RBRIGHTNESSINI    : std_logic_vector(7  downto 0);
signal   GCONTRASTINI              : std_logic_vector(7  downto 0);
signal   GBRIGHTNESSINI    : std_logic_vector(7  downto 0);
signal   BCONTRASTINI              : std_logic_vector(7  downto 0);
signal   BBRIGHTNESSINI    : std_logic_vector(7  downto 0);
-------------------------------------------------------------------------------
signal   HOST_ISUSB        : std_logic_vector(9 downto 0) ;
signal   ADDR_IN           : std_logic_vector(7 downto 0) ;
signal   DATA_IN           : std_logic_vector(7 downto 0) ;
signal   WR                : std_logic;
signal   WR_FCL            : std_logic;
signal   WR_D1             : std_logic;
signal   WR_D2             : std_logic;
signal   RD                : std_logic;
signal   VS_D1                     : std_logic;
signal   VS_D2                     : std_logic;
signal   VS_FCL                    : std_logic;
-------------------------------------------------------------------------------

begin
-------------------------------------------------------------------------------
-- initial value assignment
-------------------------------------------------------------------------------
```

sclosed To And Understood By Me

Name Date Signature:

Name Date Date: 2000/5/17

Subject: _____

17143

```
File: D:\projects\ttfcII\escalade_db\proj_zuracII\zSIM_TOP_V1\blk\sim3\host1f_gc
.vhd  2000/3/9, 05:16:14PM
-------------------------------------------------------------------------------

RCONTRASTINI     <= RCONTRAST_INI;
RBRIGHTNESSINI   <= RBRIGHTNESS_INI;
GCONTRASTINI     <= GCONTRAST_INI;
GBRIGHTNESSINI   <= GBRIGHTNESS_INI;
BCONTRASTINI     <= BCONTRAST_INI;
BBRIGHTNESSINI   <= BBRIGHTNESS_INI;
-------------------------------------------------------------------------------
-- HOST_OBUS     <= CONTRAST_OUT OR BRIGHTNESS_OUT;
-- DE_DIM register value          26/10/99
DMH_STARTINI     <= BH_START_INI;
DMH_ENDINI       <= BH_END_INI;
DMV_STARTINI     <= BV_START_INI;
DMV_ENDLINI      <= BV_END_INI(7 downto 0);
DMV_ENDHINI      <= DE_LIGHT_INI & '0' & BV_END_INI(10 downto 8);
-- CLUT_ALPHA register value          01/11/99
CLUT_ALPHAINI    <= "0000" & CLUT_ALPHA_INI;
-- BUS for DE_DIM
DE_LIGHT_DATA    <= DMV_ENDH_DATA(7 downto 4);
DMV_END_DATA     <= DMV_ENDH_DATA(2 downto 0) &  DMV_ENDL_DATA;
DEDIM_LIGHT              <= DE_LIGHT_DATA ;
DM_V_AEND                <= DMV_END_DATA ;
DM_V_ASTART      <= DMV_START_DATA ;
DM_H_AEND                <= DMH_END_DATA ;
DM_H_ASTART      <= DMH_START_DATA ;

-- CLUT_ALPHA register value          01/11/99
CLUT_ALPHA               <= CLUT_ALPHA_DATA(3 downto 0) ;
-- BUS for contrast and brightness
RCAB_CBUS(7  downto  0) <= RCONTRAST_DATA;
RCAB_CBUS(15 downto  8) <= RBRIGHTNESS_DATA;
GCAB_CBUS(7  downto  0) <= GCONTRAST_DATA;
GCAB_CBUS(15 downto  8) <= GBRIGHTNESS_DATA;
BCAB_CBUS(7  downto  0) <= BCONTRAST_DATA;
BCAB_CBUS(15 downto  8) <= BBRIGHTNESS_DATA;

LUT_CBUS(7 downto 0) <= DATA_IN;
LUT_CBUS(          8) <= LUTWADDR_write;
-- BUS for scaling
GC_CBUS(7  downto   0) <= DATA_IN;
GC_CBUS(               8) <= GRWADDR_write;
GC_CBUS(               9) <= GGWADDR_write;
GC_CBUS(              10) <= GBWADDR_write;
-------------------------------------------------------------------------------
-- Assign IBUS value
-------------------------------------------------------------------------------
ADDR_IN <= HOST_IBUS(7 downto 0) ;
DATA_IN <= HOST_IBUS(15 downto 8) ;
WR      <= HOST_IBUS(16) ;
RD      <= HOST_IBUS(17) ;


-------------------------------------------------------------------------------
p_pul_gen: process (CLK, RSTN)
begin
if (CLK'event and CLK = '1') then
    if (RSTN = '0' or RSTN = 'L' ) then
        WR_D1 <= '0' ;
        WR_D2 <= '0' ;
        WR_PUL <= '0' ;
    else
        WR_D1 <= WR ;
        WR_D2 <= WR_D1;
        WR_PUL <= WR_D1 and not(WR_D2) ;
    end if ;
end if ;
end process ;

vs_PUL_gen: PROCESS(CLK, RSTN)
  BEGIN
      IF (CLK'event and CLK = '1') THEN
        IF (RSTN = '0' or RSTN = 'L' ) THEN
          VS_D1 <= '0';
          VS_D2 <= '0';
          VS_PUL <= '0';
        ELSE
          VS_D1 <= PVS;
          VS_D2 <= VS_D1;
          VS_PUL <= VS_D1 AND NOT(VS_D2) ;
        END IF;
      END IF;
  END PROCESS;

HOST_IBUSN <= WR_PUL & RD & DATA_IN ;

-------------------------------------------------
-- CLUT_ALPHA register value          01/11/99    --
-------------------------------------------------
I_CLUT_ALPHA : DEC_REG_1ADDR -- reg B6
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
```

(RGB) CAB - CBUS

BRIGHTNESS CONTRAST

WR PUL AND GRWADDR_GN

(6, 8)

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 00/1/17

17144

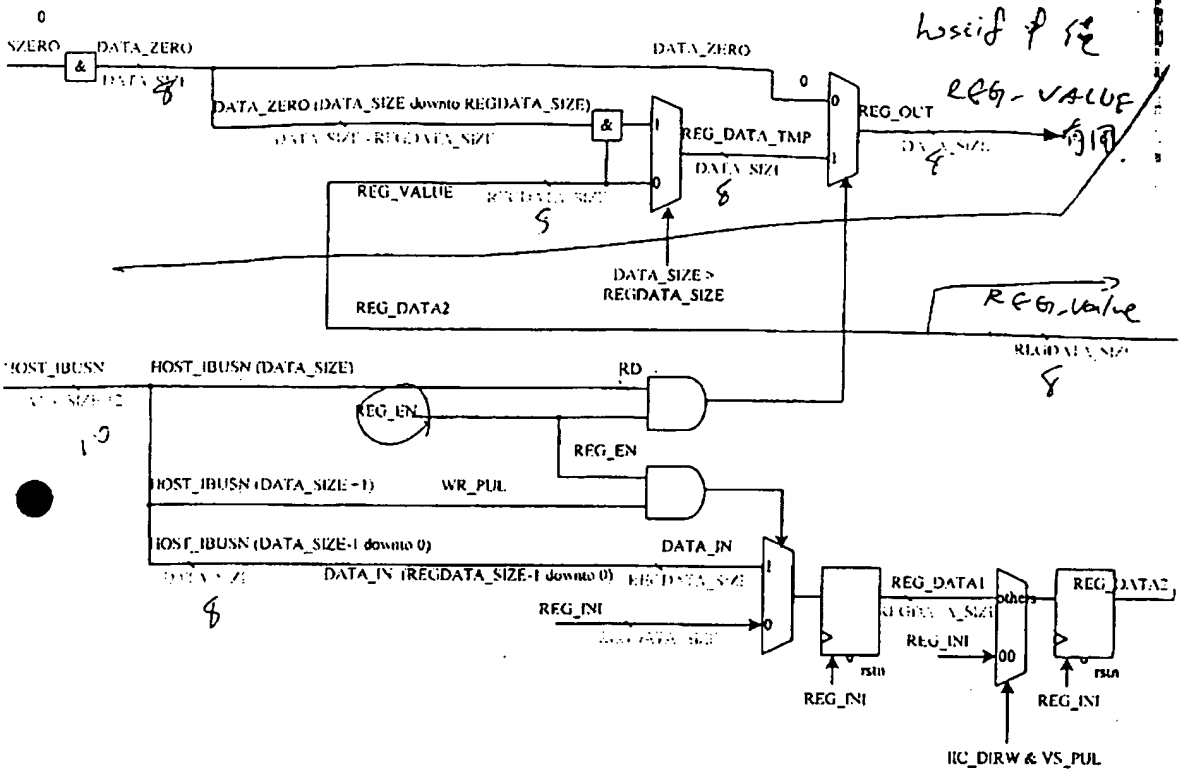
ubject: DEC_REG_1ADDR.vhd code study

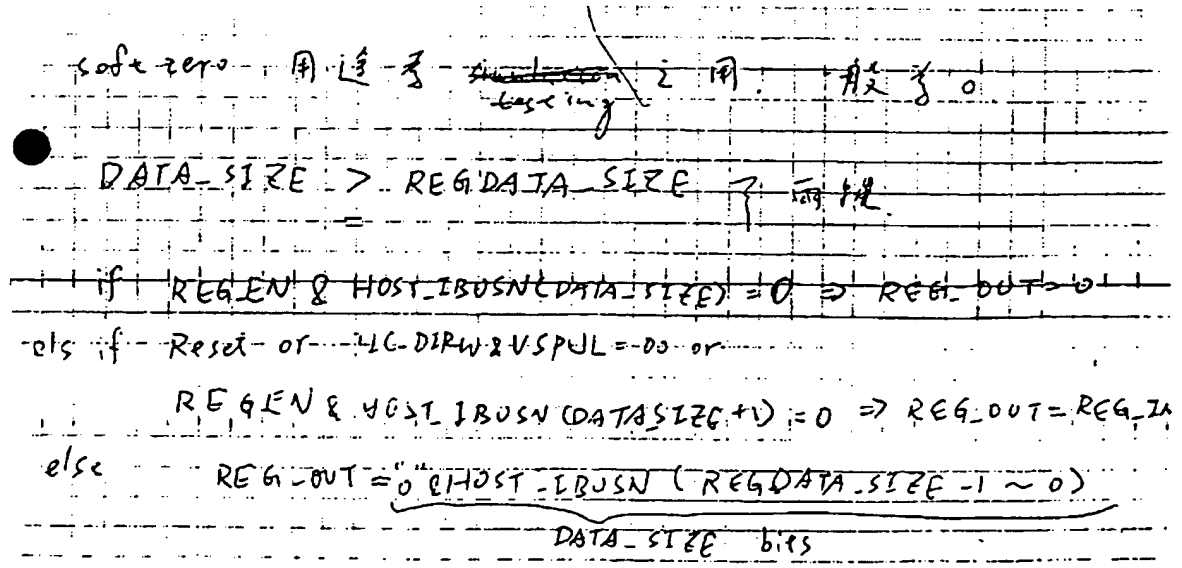
```
File: D:\projects\ttfcII\escalade_db\proj_scaler_vhdl\@CODEGEN\DEC_REG_1ADDR.vhd
  2000/3/5, 04:46:26PM
-- Author:          albert
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
library ttfc_pkgs_lib;
use ttfc_pkgs_lib.constant_pkg.all ;-- escalade ports -begin
entity DEC_REG_1ADDR is
  generic (
    ADDR_SIZE :          integer := 8;
    DATA_SIZE :          integer := 8;
    REGDATA_SIZE :       integer := 8
  );
  port (
    CLK :           in  std_logic;
    RSTN :          in  std_logic;
    SZERO :         in  std_logic;
    IIC_DIRW :      in  std_logic;
    WS_PUL :        in  std_logic;
    HOST_IBUSN :    in  std_logic_vector(ADDR_SIZE +  1 downto 0);
    REG_EN :        in  std_logic;
    REG_INI :       in  std_logic_vector(REGDATA_SIZE -  1 downto 0);
    REG_VALUE :     out std_logic_vector(REGDATA_SIZE -  1 downto 0);
    REG_OUT :       out std_logic_vector(DATA_SIZE -  1 downto 0)
  );
end DEC_REG_1ADDR;
-- escalade ports -end


architecture lan of DEC_REG_1ADDR  is

--signal REG_DATA     : std_logic_vector(REGDATA_SIZE-1 downto 0);
signal REG_DATA1    : std_logic_vector(REGDATA_SIZE-1 downto 0);
signal REG_DATA2    : std_logic_vector(REGDATA_SIZE-1 downto 0);
signal REG_DATA_TMP: std_logic_vector(DATA_SIZE-1 downto 0);
signal DATA_ZERO    : std_logic_vector(DATA_SIZE-1 downto 0);
signal DATA_IN      : std_logic_vector(7 downto 0);
signal RD           : std_logic;
signal WR_PUL       : std_logic ;

begin

REG_VALUE   <= REG_DATA2 ;
REG_DATA_TMP<= DATA_ZERO(DATA_SIZE-1 downto REGDATA_SIZE)
               & REG_DATA2 when(DATA_SIZE>REGDATA_SIZE)else REG_DATA2 ;
REG_OUT          <= REG_DATA_TMP when(RD='1' and REG_EN='1') else DATA_ZERO ;

DATA_ZERO <= (SZERO,SZERO,SZERO,SZERO,SZERO,SZERO,SZERO,SZERO) ;

DATA_IN <= HOST_IBUSN(DATA_SIZE-1 downto 0) ;
RD      <= HOST_IBUSN(DATA_SIZE) ;
WR_PUL  <= HOST_IBUSN(DATA_SIZE+1) ;
-----------------------------------------------
-- level 1 register
-----------------------------------------------
p_regdata:
process (CLK, RSTN)
begin
if (CLK'event and CLK = '1') then
        if (RSTN = '0' or RSTN = 'L' ) then
        REG_DATA1 <= REG_INI ;
    else
        if (REG_EN = '1' and WR_PUL= '1' ) then
           REG_DATA1 <= DATA_IN(REGDATA_SIZE-1 downto 0) ;
        end if ;
    end if ;
end if ;
end process ;
-----------------------------------------------
-- level 2 register
-----------------------------------------------

process (CLK, RSTN)
begin
if (CLK'event and CLK = '1') then
        if (RSTN = '0' or RSTN = 'L' ) then
        REG_DATA2 <= REG_INI ;
    else
        if (IIC_DIRW='0') then
                if (WS_PUL= '1' ) then
                   REG_DATA2 <= REG_DATA1;
                end if ;
                else
                   REG_DATA2 <= REG_DATA1;
                end if ;
    end if ;
end if ;
end process ;
end lan;
```

sclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2000/5/17

Subject: Flow chart of DEC-REG_IADDR





Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: Dec/5/18

ubject: 17146 hostif_gc study

File: D:\projects\ttfclI\escalade_ab\proj_zuracII\zSIM_TOP_V1\blk\sim3\hostif_gc.vhd 2000/3/9, 05:16:14PM

```
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN      => CLUT_ALPHA_EN,
                REG_INI     => CLUT_ALPHAINI,
                REG_VALUE => CLUT_ALPHA_DATA,
                REG_OUT     => CLUT_ALPHA_OUT );


-------------------------------------------------
-- DE_DIM register value              26/10/99   --
-------------------------------------------------
I_DMH_Start :DEC_REG_2ADDRUP -- reg 49,48
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL      => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REGM_EN     => DMH_STARTM_EN,
                REGL_EN     => DMH_STARTL_EN,
                REG_INI     => DMH_STARTINI,
                REG_VALUE => DMH_START_DATA,
                REG_OUT     => DMH_START_OUT );

I_DMH_END :DEC_REG_2ADDRUP -- reg 4B,4A
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL      => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REGM_EN     => DMH_ENDM_EN,
                REGL_EN     => DMH_ENDL_EN,
                REG_INI     => DMH_ENDINI,
                REG_VALUE => DMH_END_DATA,
                REG_OUT     => DMH_END_OUT );

I_DMV_Start :DEC_REG_2ADDRUP -- reg 4D,4C
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL      => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REGM_EN     => DMV_STARTM_EN,
                REGL_EN     => DMV_STARTL_EN,
                REG_INI     => DMV_STARTINI,
                REG_VALUE => DMV_START_DATA,
                REG_OUT     => DMV_START_OUT );

I_DMVH_END : DEC_REG_1ADDR -- reg 4F
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN      => DMV_ENDM_EN,
                REG_INI     => DMV_ENDMINI,
                REG_VALUE => DMV_ENDM_DATA,
                REG_OUT     => DMV_END_OUT );

I_DMVL_END : DEC_REG_1ADDR -- reg 4E
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN      => DMV_ENDL_EN,
                REG_INI     => DMV_ENDLINI,
                REG_VALUE => DMV_ENDL_DATA,
```

"0000" & "0000" → CLUT_ALPHA (LSB 4 bits)

→ DM_H_ASTART

→ DM_H_AEND

→ DM_V_ASTART

→ DM_V_AEND

→ DM_V_AEND

sclosed To And Understood By Me

Name	Date	Signature:
Name	Date	Date: 2000/3/13

Subject: DEC_REG_2ADDRUP vdi sendy

17147

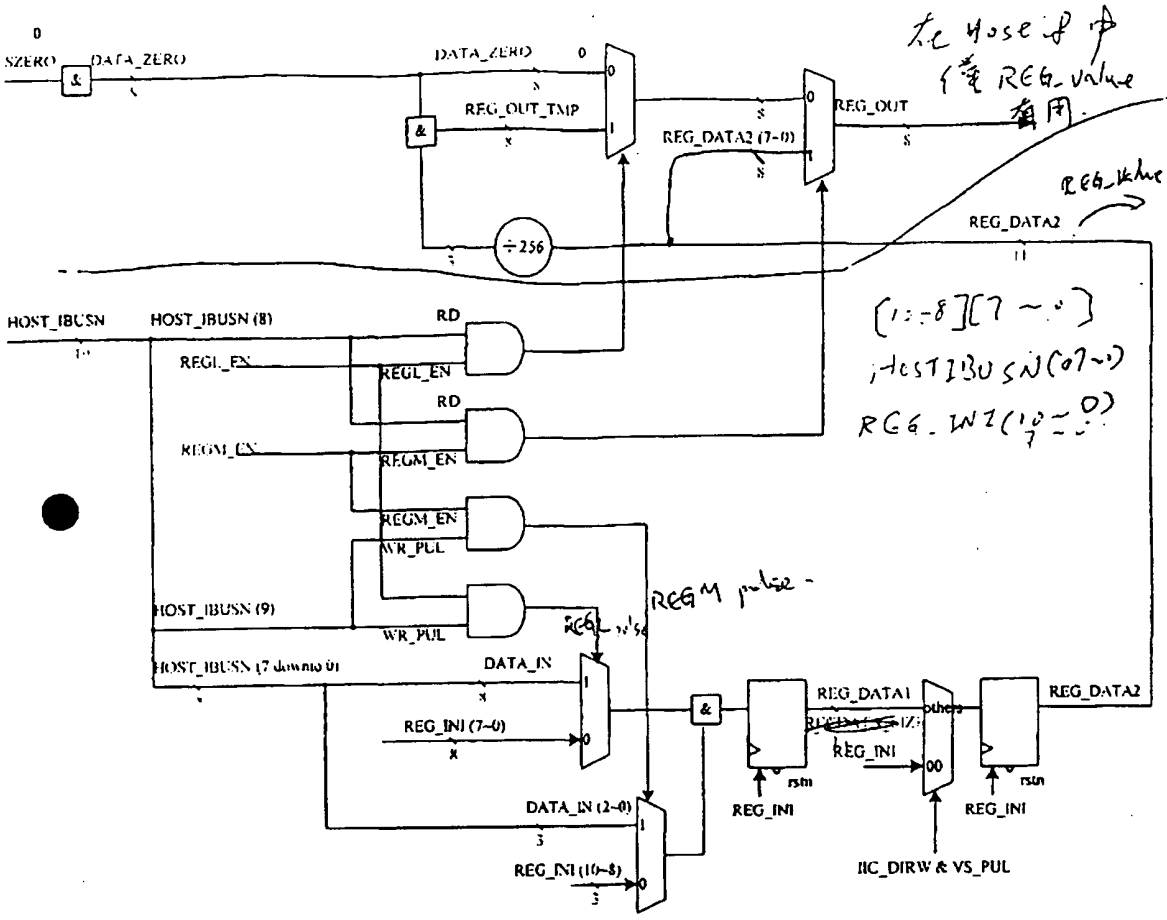
```
File: D:\projects\ttfcII\escalade_db\proj_scaler_vhdl\@CODEGEN\DEC_REG_2ADDRUP.v
hd  2000/3/5, 04:46:26PM

--  Author:             cseng
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;-- escalade ports -begin
entity DEC_REG_2ADDRUP is
  port (
    CLK :          in  std_logic;
    RSTN :         in  std_logic;
    SZERO :        in  std_logic;
    IIC_DIRW :     in  std_logic;
    VS_PUL :       in  std_logic;
    HOST_IBUSN :   in  std_logic_vector(9 downto 0);
    REGM_EN :      in  std_logic;
    REGL_EN :      in  std_logic;
    REG_INI :      in  std_logic_vector(10 downto 0);
    REG_VALUE :    out std_logic_vector(10 downto 0);
    REG_OUT :      out std_logic_vector(7 downto 0)
  );
end DEC_REG_2ADDRUP;
-- escalade ports -end


architecture lan of DEC_REG_2ADDRUP  is

signal REG_DATA1    : std_logic_vector(10 downto 0);
signal REG_DATA2    : std_logic_vector(10 downto 0);
signal ADDR_IN      : std_logic_vector(7 downto 0);
signal DATA_IN      : std_logic_vector(7 downto 0);
signal DATA_ZERO    : std_logic_vector(7 downto 0);
signal REG_OUT_TMP  : std_logic_vector(7 downto 0);
signal WR_PUL       : std_logic;
signal RD           : std_logic;

begin

REG_VALUE    <= REG_DATA2 ;
REG_OUT          <= REG_DATA2(7 downto 0) when(RD='1' and REGM_EN='1') else
                    REG_OUT_TMP                          when(RD='1' and REGL_EN='1')
else DATA_ZERO ;

DATA_ZERO <= (SZERO,SZERO,SZERO,SZERO,SZERO,SZERO,SZERO,SZERO) ;
REG_OUT_TMP <= DATA_ZERO(4 downto 0)&REG_DATA2(10 downto 8) ;

DATA_IN <= HOST_IBUSN(7 downto 0) ;
RD      <= HOST_IBUSN(8) ;
WR_PUL  <= HOST_IBUSN(9) ;
---------------------------------------
-- level 1
---------------------------------------
p_regdata:
process (CLK, RSTN)
begin
if (CLK'event and CLK = '1') then
        if (RSTN = '0' or RSTN = 'L' ) then
        REG_DATA1 <= REG_INI ;
    else
        if (REGL_EN = '1' and WR_PUL= '1' ) then
           REG_DATA1(7 downto 0) <= DATA_IN ;
        end if ;
        if (REGM_EN = '1' and WR_PUL= '1' ) then
           REG_DATA1(10 downto 8) <= DATA_IN(2 downto 0) ;
        end if ;
    end if ;
end if ;
end process ;
---------------------------------------
-- level 2
---------------------------------------
process (CLK, RSTN)
begin
if (CLK'event and CLK = '1') then
        if (RSTN = '0' or RSTN = 'L' ) then
        REG_DATA2 <= REG_INI ;
    else
        if (IIC_DIRW='0') then
                if (VS_PUL= '1' ) then
                   REG_DATA2 <= REG_DATA1;
                end if ;
                else
                   REG_DATA2 <= REG_DATA1;
                end if ;
    end if ;
end if ;
end process ;

end lan;
```

sclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2000/5/18

Subject: 17148 DEC_REG_2ADDRUP flow chart



if HOST_IBUSN(8)='0' or REGL_EN = REGM_EN = '0' => REG_OUT='0'

elsif rsen or IIC_DIRW & VS_PUL = '0' or WR_PUL='0' or REGM_EN=REGL_EN='0'

if RD & REGM_EN = '1' => REG_out = REG_IN (7~0)

elsif RD & REGL_EN = '1' => REG_out = REG_IN

end if

else

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2020/5/18

Subject:

1. XWR_PUL = 0 => REG_DATA2 = REG_IN1
   rsen = 1
   IC_DIRW = 0 AND VS_PUL = 0
   REGM_EN = REGL_EN = 0

2. if WR_PUL = 1
   ① REGM = REGL = 1 => REG_DATA1 = DATA_IN (7~0, 7~0)
   ② REGM = 1, REGL = 0 => REG_DATA1 = DATA_IN (7~0) & REG_IN1 (7~
   ③ REGM = 0, REGL = 1 => REG_DATA1 = REG_IN1 (15~8) & DATA_IN

3. if RD = 1
   ① REGM = REGL = 1 => ~~REG_DA~~ REG_OUT = REG ~~DATA~~ 2 (7~
   may = DATA_IN (7~0)
   ② REGM = 1, REGL = 0 => REG_OUT = REG DATA2 (7~0)
   may = REG_IN1 (7~0)
   ③ REGM = 0, REGL = 1 => REG_OUT = REG_~~IN1~~ DATA2 (15~8)
   may = REG_IN1 / 256

M: MSB
L: LSB

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: ~~/5/19

17150

Subject: _____

File: D:\projects\ttfcII\escalade_db\proj_zuracI:\zSIM_TOP_V1\nik\sim3\hostif_gc
.vhd 2000/3/9, 05:16:14PM

```
                REG_OUT    => DMV_END_OUT );

-------------------------------------------------------------------------
-- brightness & contrast register                                       --
-------------------------------------------------------------------------
IR_CONTRAST : DEC_REG_1ADDR
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK        => CLK ,
                RSTN       => RSTN,
                SZERO      => SZERO,
                IIC_DIRW   => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN     => RCONTRAST_EN,
                REG_INI    => RCONTRASTINI,
                REG_VALUE  => RCONTRAST_DATA,
                REG_OUT    => RCONTRAST_OUT      );

IR_BRIGHTNESS : DEC_REG_1ADDR
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK        => CLK ,
                RSTN       => RSTN,
                SZERO      => SZERO,
                IIC_DIRW   => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN     => RBRIGHTNESS_EN,
                REG_INI    => RBRIGHTNESSINI,
                REG_VALUE  => RBRIGHTNESS_DATA,
                REG_OUT    => RBRIGHTNESS_OUT    );

IG_CONTRAST : DEC_REG_1ADDR
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK        => CLK ,
                RSTN       => RSTN,
                SZERO      => SZERO,
                IIC_DIRW   => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN     => GCONTRAST_EN,
                REG_INI    => GCONTRASTINI,
                REG_VALUE  => GCONTRAST_DATA,
                REG_OUT    => GCONTRAST_OUT      );

IG_BRIGHTNESS : DEC_REG_1ADDR
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK        => CLK ,
                RSTN       => RSTN,
                SZERO      => SZERO,
                IIC_DIRW   => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN     => GBRIGHTNESS_EN,
                REG_INI    => GBRIGHTNESSINI,
                REG_VALUE  => GBRIGHTNESS_DATA,
                REG_OUT    => GBRIGHTNESS_OUT    );

IB_CONTRAST : DEC_REG_1ADDR
generic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK        => CLK ,
                RSTN       => RSTN,
                SZERO      => SZERO,
                IIC_DIRW   => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN     => BCONTRAST_EN,
                REG_INI    => BCONTRASTINI,
                REG_VALUE  => BCONTRAST_DATA,
                REG_OUT    => BCONTRAST_OUT      );

IB_BRIGHTNESS : DEC_REG_1ADDR
```

R CONTRAST

R-BRIGHTNESS

G - CONTRAST

G-BRIGHTNESS

B-CONTRAST

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/3/13

Subject: _____

```
File: D:\projects\tticII\escalade_db\proj_zuraciI\zSIM_TOP_V1\blk\sim3\hostif_gc
.vhd  2000/3/9, 05:16:14PM

genecic map (
                ADDR_SIZE => 8, -- address bits size
                DATA_SIZE => 8, -- data transfer bits size
        REGDATA_SIZE => 8) -- reg used bits size
port map(
                CLK         => CLK ,
                RSTN        => RSTN,
                SZERO       => SZERO,
                IIC_DIRW    => IIC_DIRW,
                VS_PUL   => VS_PUL,
                HOST_IBUSN=> HOST_IBUSN,
                REG_EN      => BBRIGHTNESS_EN,
                REG_INI     => BBRIGHTNESSINI,
                REG_VALUE => BBRIGHTNESS_DATA,
                REG_OUT     => BBRIGHTNESS_OUT    );

GRWADDR_write <= WR_PUL AND GRWADDR_EN;
GGWADDR_write <= WR_PUL AND GGWADDR_EN;
GBWADDR_write <= WR_PUL AND GBWADDR_EN;
LUTWADDR_write <= WR_PUL AND LUTWADDR_EN;

-------------------------------------------------------------------------------
-- Address decoder
-------------------------------------------------------------------------------
adec: PROCESS(ADDR_IN)
BEGIN
--  register
RCONTRAST_EN      <= '0'; -- reg AE
RBRIGHTNESS_EN    <= '0'; -- reg AF
GCONTRAST_EN      <= '0'; -- reg B0
GBRIGHTNESS_EN    <= '0'; -- reg B1
BCONTRAST_EN      <= '0'; -- reg B7
BBRIGHTNESS_EN    <= '0'; -- reg B8
GRWADDR_EN        <= '0'; -- reg B2
GGWADDR_EN        <= '0'; -- reg B3
GBWADDR_EN        <= '0'; -- reg B4
LUTWADDR_EN       <= '0'; -- reg B5
-- DE DIM register
DMH_STARTL_EN     <= '0';
DMH_STARTM_EN     <= '0';
DMH_ENDL_EN       <= '0';
DMH_ENDM_EN       <= '0';
DMV_STARTL_EN     <= '0';
DMV_STARTM_EN     <= '0';
DMV_ENDL_EN       <= '0';
DMV_ENDM_EN       <= '0';
-- CLUT_ALPHA register
CLUT_ALPHA_EN     <= '0';

CASE ADDR_IN IS
-- contrast and brightness register
WHEN RCONTRAST_ADDR                 => RCONTRAST_EN <= '1';    -- reg AE
WHEN RBRIGHTNESS_ADDR     => RBRIGHTNESS_EN <= '1'; -- reg AF
WHEN GCONTRAST_ADDR                 => GCONTRAST_EN <= '1';    -- reg B0
WHEN GBRIGHTNESS_ADDR     => GBRIGHTNESS_EN <= '1'; -- reg B1
WHEN BCONTRAST_ADDR                 => BCONTRAST_EN <= '1';    -- reg B7
WHEN BBRIGHTNESS_ADDR     => BBRIGHTNESS_EN <= '1'; -- reg B8
WHEN GRWADDR_ADDR                   => GRWADDR_EN <= '1';      -- reg B2
WHEN GGWADDR_ADDR                   => GGWADDR_EN <= '1';      -- reg B3
WHEN GBWADDR_ADDR                   => GBWADDR_EN <= '1';      -- reg B4
WHEN LUTWADDR_ADDR                  => LUTWADDR_EN <= '1';     -- reg B5
-- DE DIM register
WHEN DMH_STARTL_ADDR      => DMH_STARTL_EN <= '1';
WHEN DMH_STARTM_ADDR      => DMH_STARTM_EN <= '1';
WHEN DMH_ENDL_ADDR        => DMH_ENDL_EN <= '1';
WHEN DMH_ENDM_ADDR        => DMH_ENDM_EN <= '1';
WHEN DMV_STARTL_ADDR      => DMV_STARTL_EN <= '1';
WHEN DMV_STARTM_ADDR      => DMV_STARTM_EN <= '1';
WHEN DMV_ENDL_ADDR        => DMV_ENDL_EN <= '1';
WHEN DMV_ENDM_ADDR        => DMV_ENDM_EN <= '1';
-- CLUT_ALPHA register
WHEN CLUT_ALPHA_ADDR      => CLUT_ALPHA_EN <= '1';
--
WHEN OTHERS               => NULL;
END CASE;
END PROCESS adec;

end behavioral ;
```

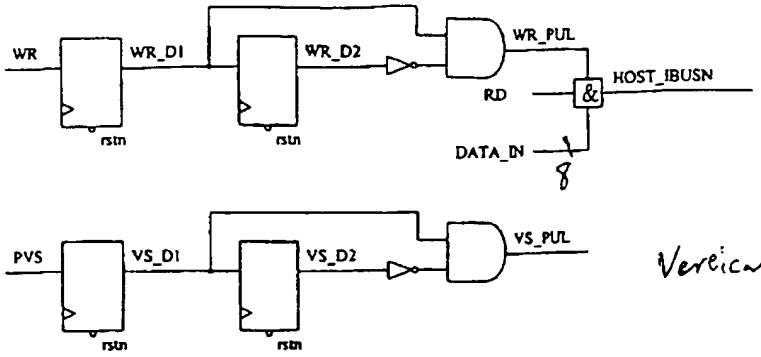
B-BRIGHTNESS

Disclosed To And Understood By Me

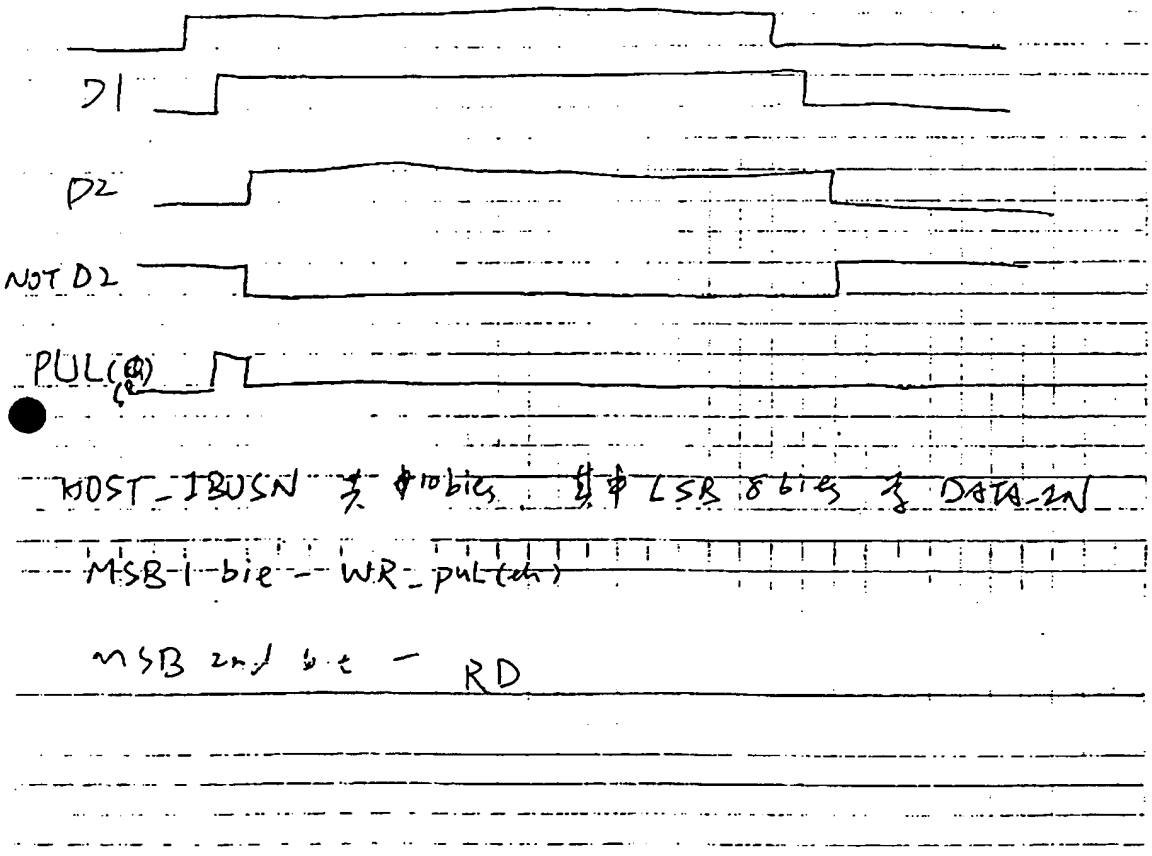
Name	Date	Signature: Blair
Name	Date	Date: 2000/3/19

17152

Subject: _____



Vertical scan pulse



HOST_IBUSN → 10 bits, → LSB 8 bits → DATA_IN

MSB-1-bit → WR_pul(ch)

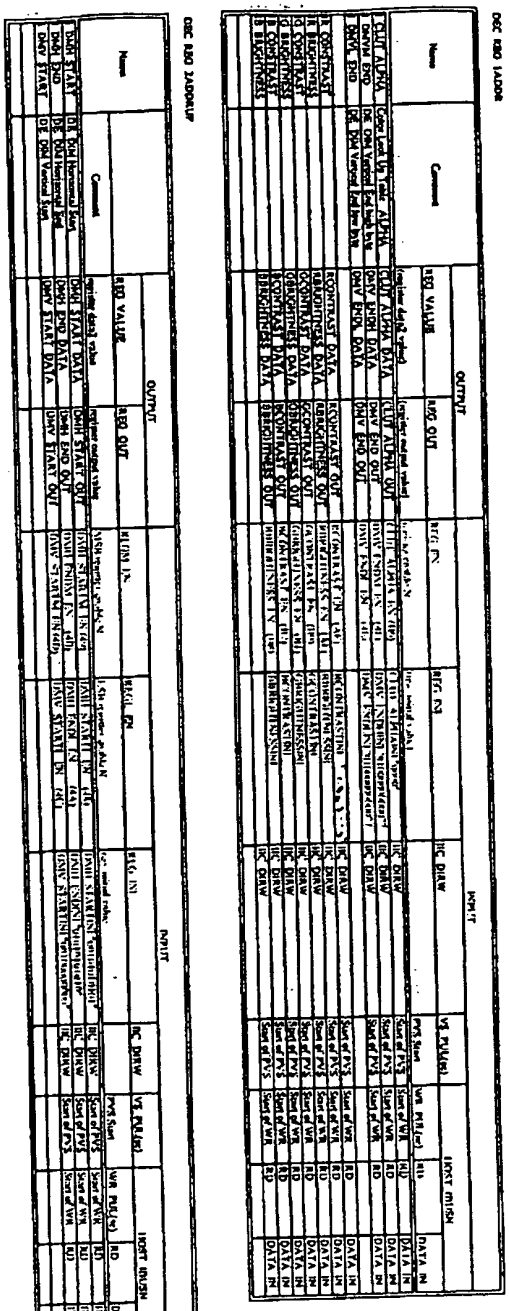
MSB 2nd bit → RD

Disclosed To And Understood By Me

Name	Date	Signature: olsiv
Name	Date	Date: 11/19

17153

Subject: _____



Disclosed To And Understood By Me

_____ Signature: _____
Name Date

_____ Date: _____
Name Date

17156

Subject: SRAM 256 x 10

File: D:\projects\ttfcII\escalade_db\proj_zuracII\@CODEGEN\sram_256x10.vhd 2000/1/31, 08:28:06PM

```
-- Copyright (c) 1994-1999 Escalade Corporation. All rights reserved.
-- Block proj_zuracII/sram_256x10:behavioral
--   10 inputs
--   2 outputs
--   0 inouts
--
-- RTL code generated by DesignBook LAN
--
-- Title:          proj_zuracII/sram_256x10:behavioral
-- Author:         albert
-- Date:           12/21/99 19:43
-- Description:
--
--
--

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
library ttfc_pkgs_lib;
use ttfc_pkgs_lib.constant_pkg.all ;
use ttfc_pkgs_lib.reg_table_pkg.all ;

-- pragma translate_off
--library MATRIX_LIB ;
library TIC_Lib ;
library FBC_Lib ;
library [illegible]
library [illegible]
library tsmc_io_025_lib;
-- library ttfc_gate_lib ;
-- use ttfc_gate_lib.all ;
-- pragma translate_on

-- escalade [illegible]

---------------------------------------------------------------------
--
-- File        : sram_256x10.vhdl
--
-- Author(s)   : Albert Wang
--
-- Description: add test circuit into Artisan sram
--              twen is the serial input, tq(7) is the serial output
--              chain order is
--              twen -> toen -> tcen -> ta(7) -> ta(6) -> .... -> ta(0) ->
--              td(0) -> td(1) -> ... -> td(7)
--              here, asq(i) connect to ta(i-1)
--                    asq(0) connect to td(0)
--                    q(i)   connect to td(i+1)
--
-- Date        : Oct. 25 1999
--
-- Version     : 2.0
--
-- History     :
---------------------------------------------------------------------
library ieee;

use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
-- pragma translate_off
library tsmc_macro_025_lib ;
-- pragma translate_on

-- escalade ports -begin
entity sram_256x10 is
    port (in1    :    in    std_logic_vector(9 downto 0);
          clk    :    in    std_logic;
          cen    :    in    std_logic;
          oen    :    in    std_logic;
          wen    :    in    std_logic;
          a      :    in    std_logic_vector(7 downto 0);
          tis    :    in    std_logic;
          tms    :    in    std_logic;
          tqoen  :    in    std_logic;
          serial_in[illegible]
          out1   :    out   std_logic_vector(9 downto 0);
          serial_[illegible]
end sram_256x10;
-- escalade ports -end

architecture behavioral of sram_256x10 is

component sram256x10
    port (Q :    out   std_logic_vector(9 downto 0);
          CENSQ :    out   std_logic;
          OENSQ :    out   std_logic;
          WENSQ :    out   std_logic;
          ASQ : out  std_logic_vector(7 downto 0);
          TQ : out   std_logic_vector(9 downto 0);
```

Disclosed To And Understood By Me

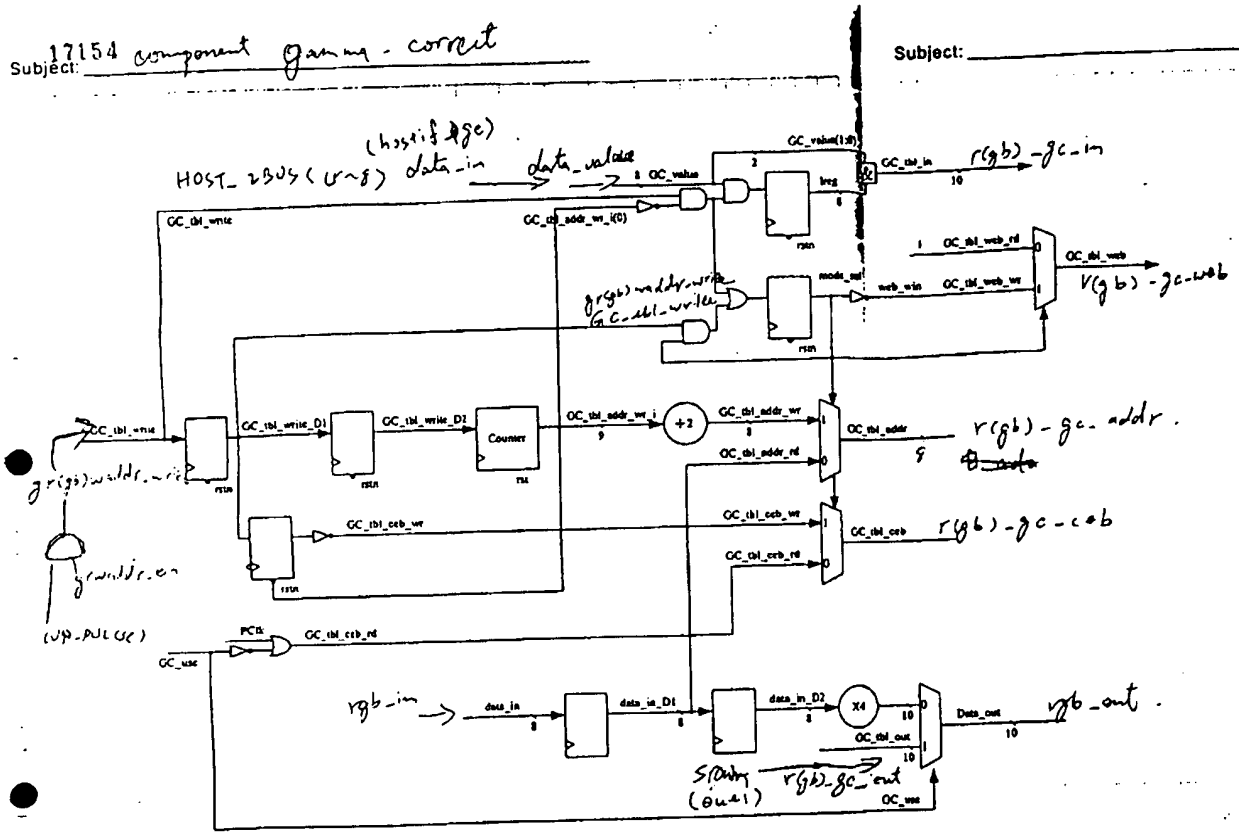
Name	Date
Name	Date

Signature:

Date:

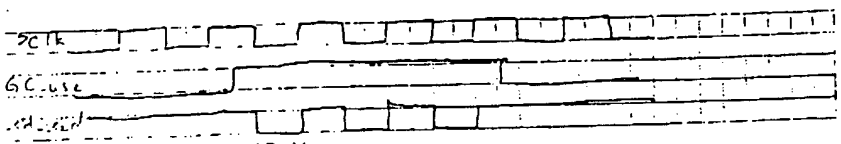
TRUMPION CONFIDENTIAL BUSINESS INFORMATION, SUBJECT TO PROTECTIVE ORDER

Subject: 17154 component gamma - correct




```
if GC_use = '1'
=> Data_out <= sram_out
   = r(gb)_gc_out
else
if GC = use => 0
=> Data_out = RGB_in
Data_out -> RGB_out
rgb_in => data_in => decode
address -[r(gb)_gc_waddr]
=> sram_out[1] -> r(gb)_gc_out
=> rgb_out
address => write
GC_tbl_in
=data_in
=HOST_BUS(10~0)
```

PCIk GC_tbl_clk -> rgb_gc_clk.



GC_tbl_write = '0' => mode_sel = 0 <=> read

GC_tbl_write = '1' => mode_sel = (up) => write
probably

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2001/5/29

Subject: 17155

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2001/5/29

Subject: _____ 17157

File: D:\projects\ttfcII\escalade_db\proj_surfcII\@CODEGEN\sram_256x10.vhd 2000/1/31, 08:28:06PM

```
            CLK : in     std_logic;
            CEN : in     std_logic;
            OEN : in     std_logic;
            WEN : in     std_logic;
            A :   in     std_logic_vector(7 downto 0);
            D :   in     std_logic_vector(9 downto 0);
            TIS : in     std_logic;
            TMS : in     std_logic;
            TCEN :         in     std_logic;
            TOEN :         in     std_logic;
            TQOEN :        in     std_logic;
            TWEN :         in     std_logic;
            TA :  in     std_logic_vector(7 downto 0);
            TD :  in     std_logic_vector(9 downto 0));
end component;

signal tq                : std_logic_vector(9 downto 0);
signal q                 : std_logic_vector(9 downto 0);
signal ta                : std_logic_vector(7 downto 0);
signal td                : std_logic_vector(9 downto 0);
signal asq               : std_logic_vector(7 downto 0);
signal twen              : std_logic;
signal tcen              : std_logic;
signal toen              : std_logic;
signal wensq     : std_logic;
signal censq     : std_logic;
signal oensq     : std_logic;

begin

twen <= serial_in;
toen <= wensq;
tcen <= oensq;
ta(7) <= censq;
ta(6 downto 0) <= asq(7 downto 1);
td(0) <= asq(0);
td(9 downto 1) <= tq(8 downto 0);
serial_out <= tq(9);


i_256x10: sram256x10
  port map
       (
        Q                => q,              -- internal use
                 CENSQ   => censq,          -- internal use
                 OENSQ   => oensq,          -- internal use
                 WENSQ   => wensq,          -- internal use
                 ASQ                => asq,          -- internal use
                 TQ                 => tq,           -- internal use
                 CLK                => clk,          -- from external
                 CEN                => cen,          -- from external
                 OEN                => oen,          -- from external
                 WEN                => wen,          -- from external
                 A                  => a,            -- from external
                 D                  => ini,          -- from external
                 TIS                => tis,          -- from external
                 TMS                => tms,          -- from external
                 TCEN    => tcen,           -- internal use
                 TOEN    => toen,           -- internal use
                 TQOEN   => tqoen,          -- from external
                 TWEN    => twen,           -- internal use
                 TA                 => ta,           -- internal use
                 TD                 => td            -- internal use
       );

outl <= q;

end behavioral;
```

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2000/5/10

17158

Subject: de-interlace Algorithm

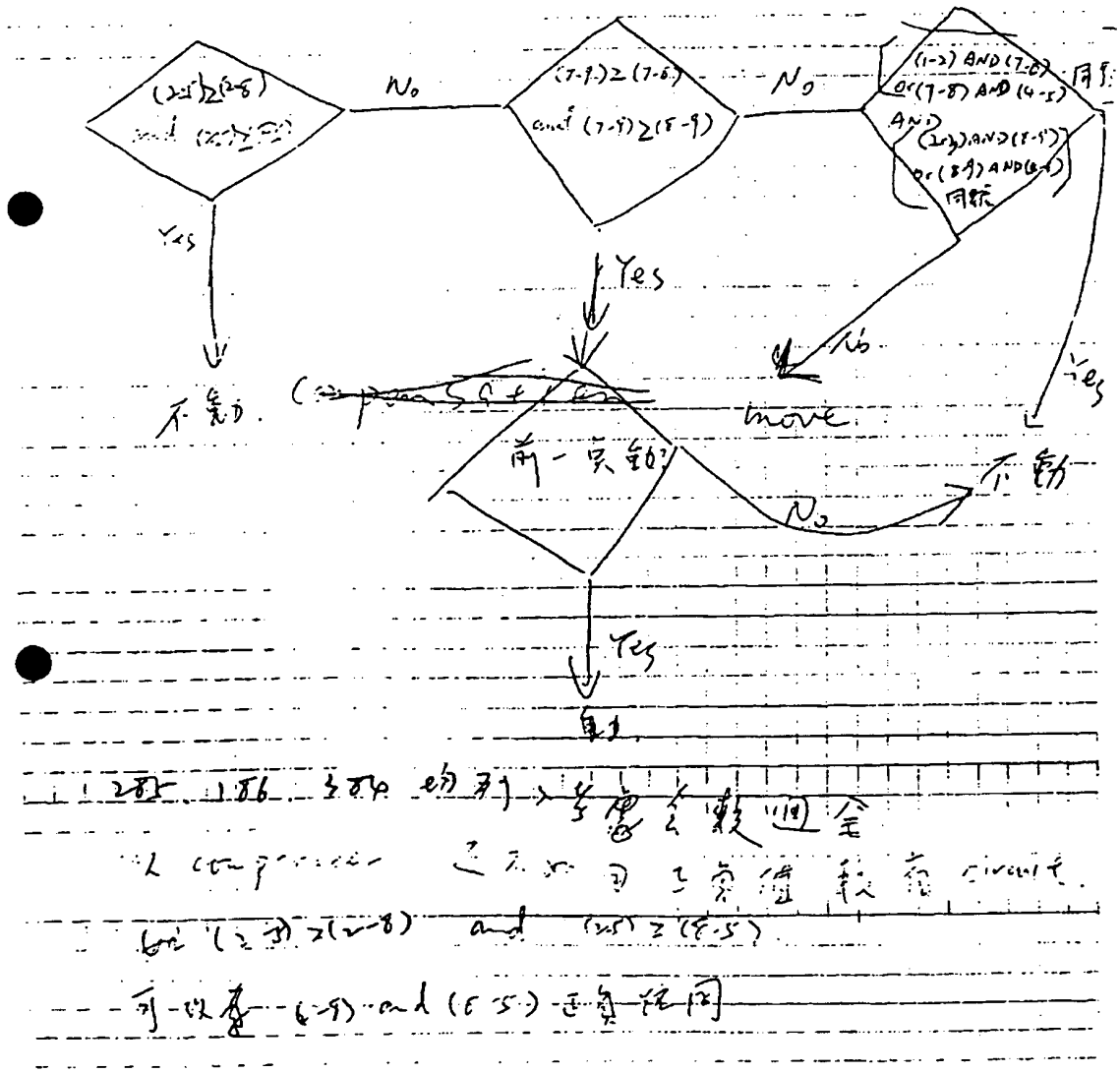
1 2 3
7 8 9
4 5 6

old ... 1 6 > threshold
... 8 2 > threshold
... 9 4 > threshold

=> motion

=> 1 6 / 2 or 2 0 1 / 2 or 1 7 2 / 2

New thoughts:



sclosed To And Understood By Me

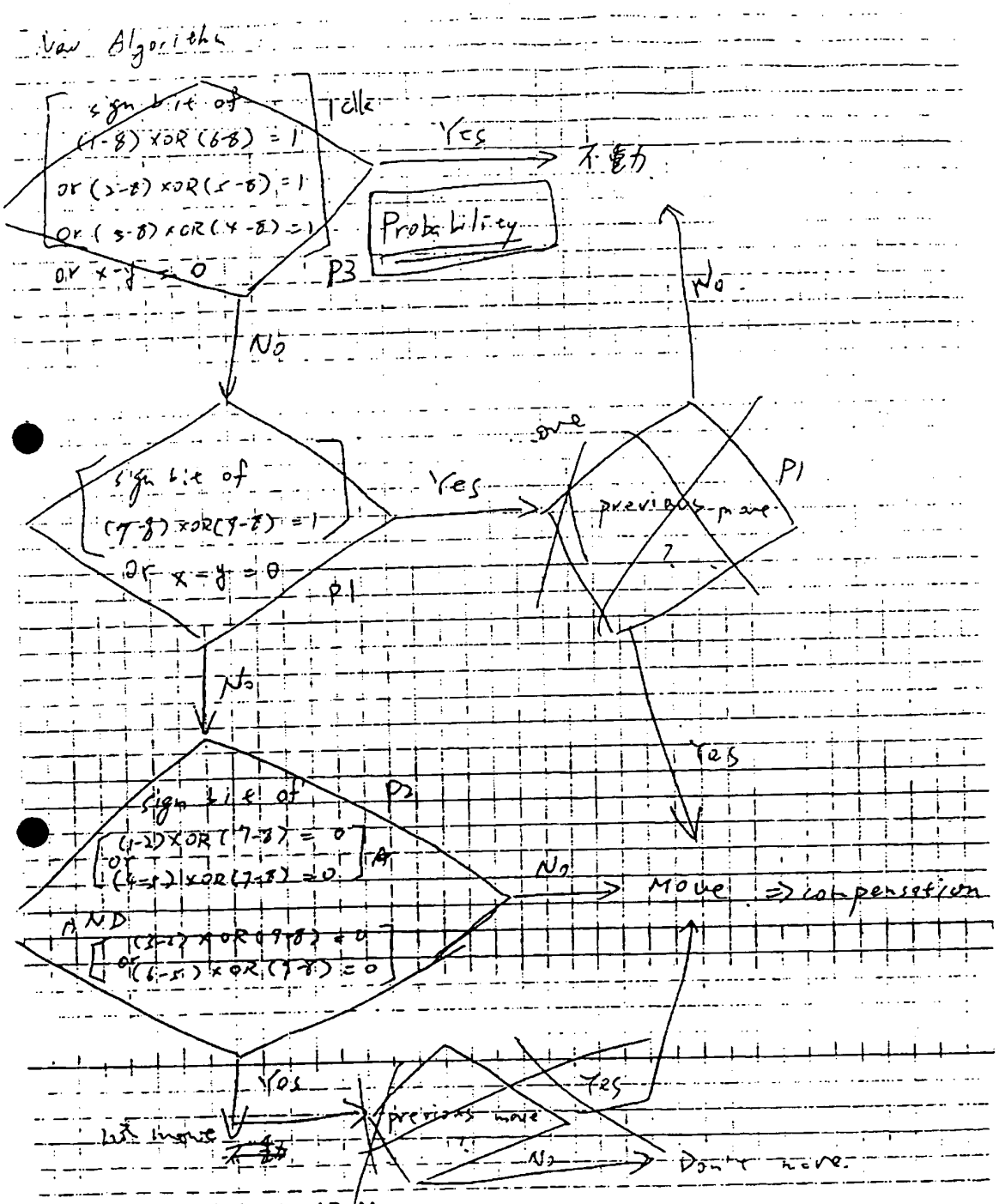
Name Date

Name Date

Signature:

Date:

Subject: _____



Disclosed To And Understood By Me

_____ _____
Name Date

_____ _____
Name Date

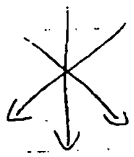
Signature: blair

Date: 2000/5/31

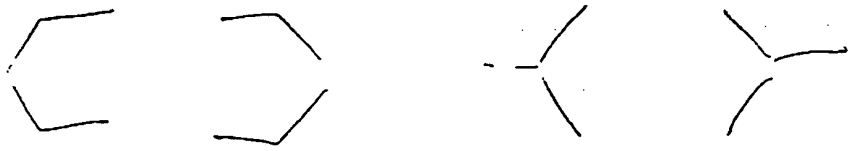
17160

Subject:

1 2 3
7 8 9
4 6



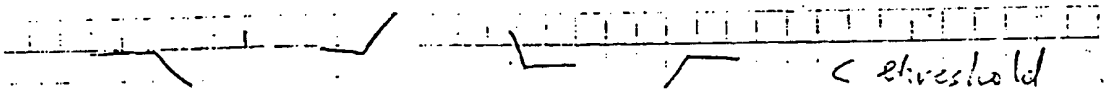
圖像可由四方、三角、圓形等構成



if 7 7 8 ≥ 9 ⇒ 7 2 5 ≥ 9 成立等不成立
或 7 2 2 9

if 7 ≤ 8 ≤ 9 ⇒ 7 ≤ 5 ≤ 9 不成立
or 7 ≤ 2 ≤ 9

if 7 7 8 < 9



⇒ 有動作 ⇒ wi

Disclosed To And Understood By Me

Name Date

Name Date

Signature:

Date: 16/1

Subject: oslo deinterlace

檔案: C:\projects\scaler\oslo\deinterlace\source code\VHDL\01\deintp2.vhdl 2000/6/21, 01:26:57PM

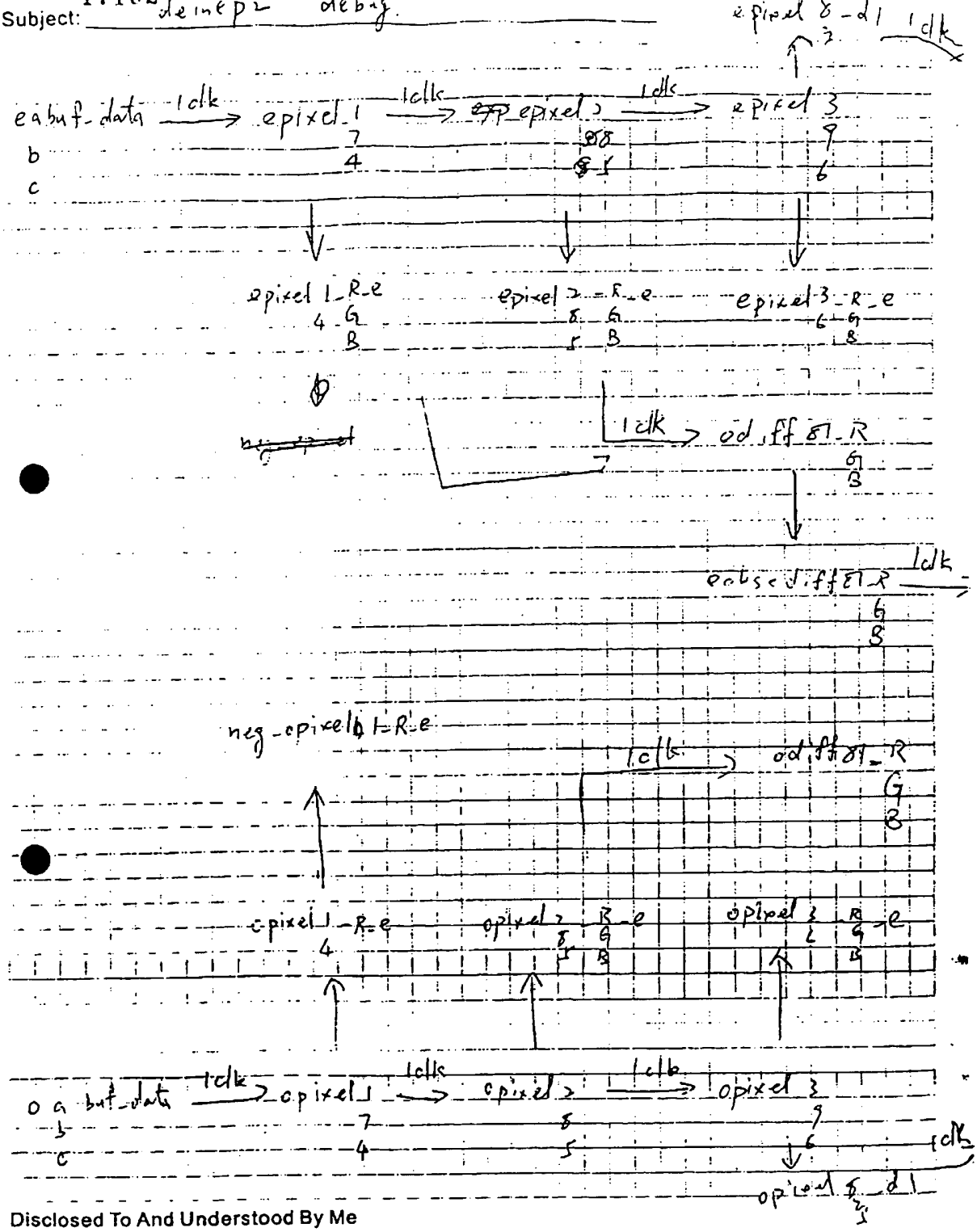
```
--------------------------------------------------------------------------------
LIBRARY ieee;

USE ieee.std_logic_1164.all;
--USE ieee.std_logic_unsigned.all;

--      Gene Chuang, Mar 30, 2000, trumpion us~
--      name: deintp2.vhdl (deinterlacing circuit for video)

--      Modified:
--        Jun 14 2000          =1 => disable
--                  "disable" (bypass the deint) signal is added,
--                  THDmotion changed from 8-bit to 11-bit
--        Jun 17 2000
--                  Modify to an even/odd structure (tandem) for
--                  OSLO. Another 3 in and 3 out buses are added.
--

--      For even and odd ports
--      input: Abuf_data (n-2 Frame t)          output:  LA_out (n-2 Frame t)
--             Bbuf_data (n-1 Frame t-1)                 LB_out (n-1 Frame t)
--             Cbuf_data (n   Frame t)                   LC_out (n   Frame t)
--
--
--      The deinterlacing ckt genearates the 3 lines (n-2, n-1, n) for scaler
--
--
--      There are 3x3 pixels for motion detection and I to Progressive
conversion,
--      namely,
--      p1 = RGB(pA,delay1); p2 = RGB(pA,delay2); p3 = RGB(pA,delay3);
--                           p8 = RGB(pB,delay2);
--      p4 = RGB(pC,delay1); p5 = RGB(pC,delay2); p6 = RGB(pC,delay3);
--
--
--      (Diff81+Diff86 && Diff82+Diff85 && Diff83+Diff84 > THDmotion) means
Motion
--      In these cases -
--      Motion: if (Diff82+Diff85 < Diff81+Diff86 && Diff82+Diff85 <
Diff83+Diff84)
--                then the result is the averaged of pixel2 and 5.
--                (in the case diff186 is the smallest, the result is the
--                 average of pixel 1 and pixel 6)
--      Stationary: the result is from the previous field (F t-1).
--                (In a sense, stationary is the same as deint disabled)
--      Since PH_active (or swap_hs), powerdown, and PV_active are not used, to
save
--      power, the input is set to zero during display disable period.
--
--         e------------e  These 3 col are for even port deinterlacing
--          1/4 2/5 3/6
--      ep1 op1 ep2 op2 ep3 op3
--      ep7 op7 ep8 op8 ep9 op9
--      ep4 op4 ep5 op5 ep6 op6
--              1/4 2/5 3/6
--             o------------o  These 3 col are for odd port
--------------------------------------------------------------------------------
entity deintp2 is
  port (
    rstn :       in  std_logic;
    clk :        in  std_logic;
    aebuf_data : in  std_logic_vector(23 downto 0);
    abbuf_data : in  std_logic_vector(23 downto 0);
    acbuf_data : in  std_logic_vector(23 downto 0);
    oabuf_data : in  std_logic_vector(23 downto 0);
    obbuf_data : in  std_logic_vector(23 downto 0);
    ocbuf_data : in  std_logic_vector(23 downto 0);
    disable    :   in std_logic;
    THDmotion  : in  std_logic_vector(10 downto 0);
    eLA_out :    out std_logic_vector(23 downto 0);
    eLB_out :    out std_logic_vector(23 downto 0);
    eLC_out :    out std_logic_vector(23 downto 0);
    oLA_out :    out std_logic_vector(23 downto 0);
    oLB_out :    out std_logic_vector(23 downto 0);
    oLC_out :    out std_logic_vector(23 downto 0)

  );
end deintp2;


architecture behavioral of deintp2 is

  component add9
  port
    (
      in1       : in  std_logic_vector(8 downto 0);
      in2       : in  std_logic_vector(8 downto 0);
      clk       : in  std_logic;
      sum       : out std_logic_vector(8 downto 0)
    );
  end component;
```

Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2000/6/21

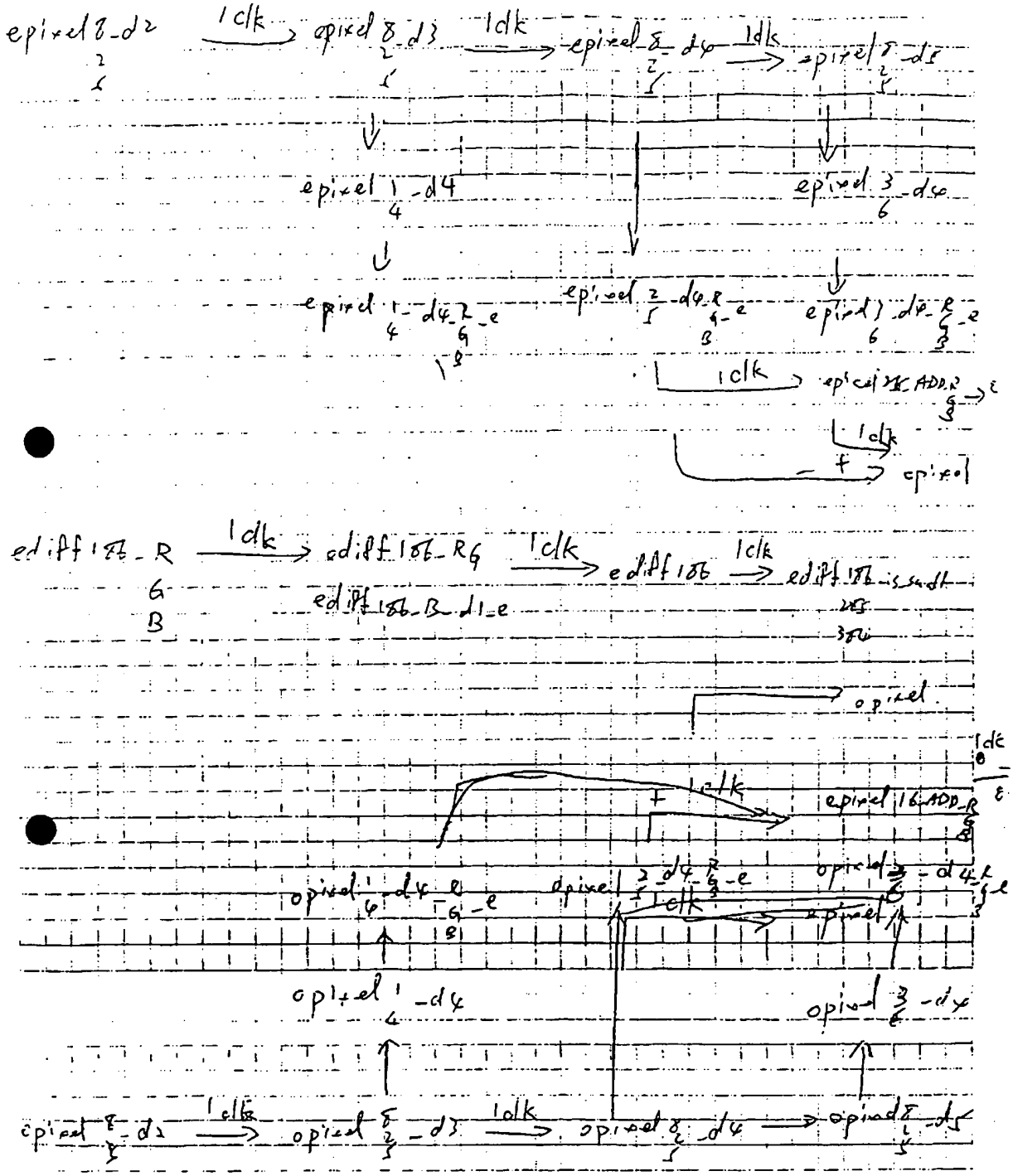
Subject: 17162 deinepr debug.



Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2002/6/13

Subject: _____



Disclosed To And Understood By Me

Name	Date	Signature: Blair
Name	Date	Date: 2003/6/23

17164

Subject:

first clk → a0 → ep1

a1 → op1

b1 → ep7

b1 → op7

c0 → ep4

c1 → op4

second clk → a2 → ep1 a0 → ep2

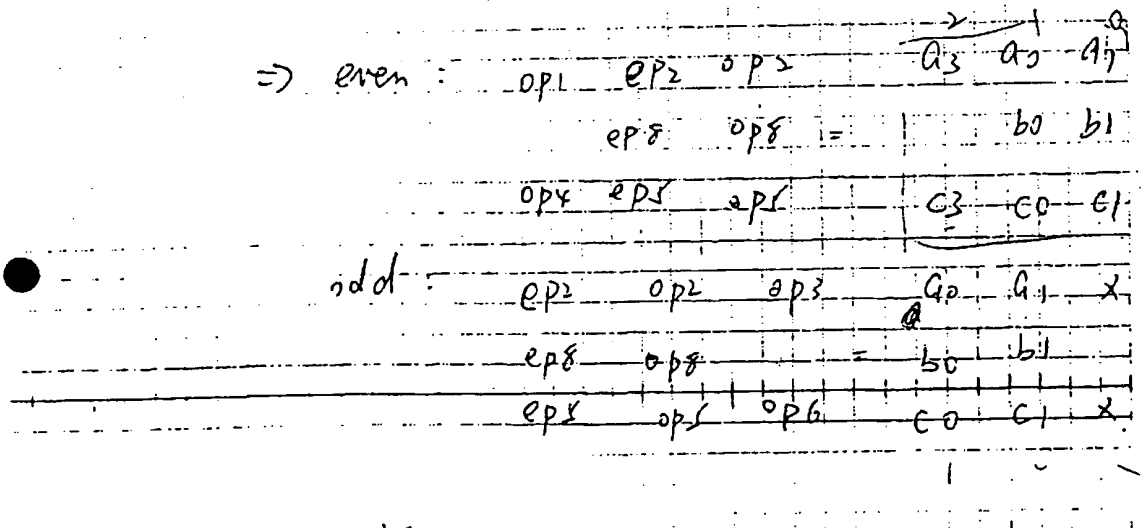
a3 → op1 a1 → op2

b2 → ep7 b0 → ep8

b3 → op7 b1 → op8

c2 → ep4 c0 → ep5

c3 → op4 c1 → op5



=> even wrong

Disclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2001/6/27

Subject: _____

third clk =>

$q_4 \rightarrow ep1$	$q_2 \rightarrow ep2$	$q_0 \rightarrow ep3$
$q_5 \rightarrow op1$	$q_3 \rightarrow op2$	$q_1 \rightarrow op1$
$b_4 \rightarrow ep7$	$b_5 \rightarrow ep8$	$b_0 \rightarrow ep9$
$b_5 \rightarrow op7$	$b_3 \rightarrow op8$	$b_1 \rightarrow op9$
$c_4 \rightarrow ep4$	$c_2 \rightarrow ep5$	$c_0 \rightarrow ep6$
$c_5 \rightarrow op4$	$c_3 \rightarrow op5$	$c_1 \rightarrow op6$

=> even : op1 ep2 op2 = q_5 q_3 q_1 (4 3 2)

op7 ep8 op8 = b_5 b_3 b_1

op4 ep5 op5 = c_5 c_3 c_1

odd : ep2 op1 ep3 = q_2 q_3 q_0 (3 2 1)

ep8 op8 ep9 = b_2 b_3 b_0

op5 op5 ep6 = c_2 c_3 c_0

=> even odd

Disclosed To And Understood By Me

Name Date

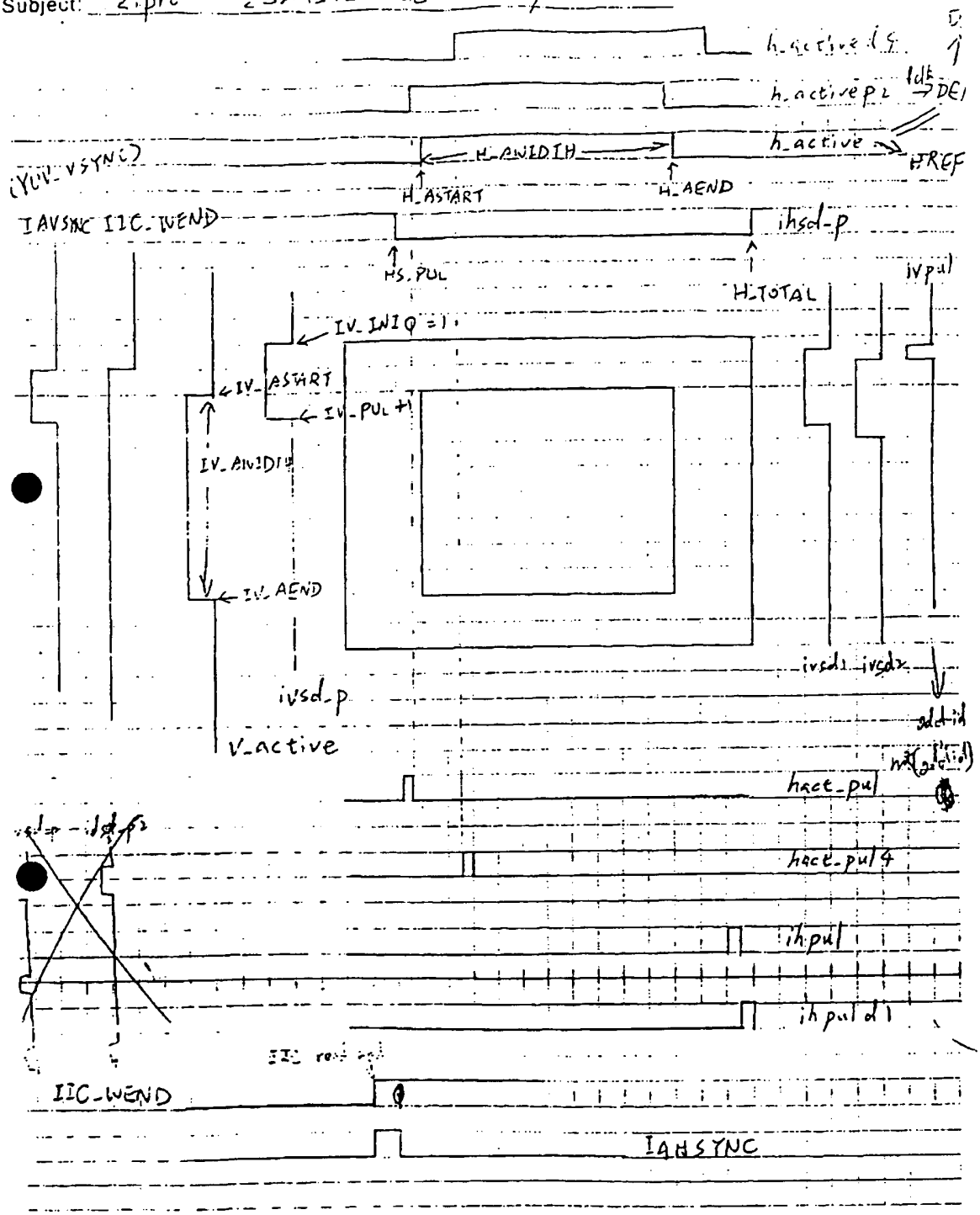
Name Date

Signature: Blair

Date: 2000/6/27

17106

Subject: zipro zSIM_VEC_GEN study



Disclosed To And Understood By Me

Name	Date	Signature: bkir
Name	Date	Date: 12-17-98

Subject: test.vec study (SVGA to XGA) 17167

```
-----------------------------------------------------------
-- Input parameter
-----------------------------------------------------------
-- HS
       11   00    0
  10   d8   0
--    1098 76543210   IH_ASTART - 0d8 H = 216 D   -> Hor Sync Time + H Back Porch
--     |||  ||||||||
--     |||  |||||||+--- 0
--     |||  ||||||+---- 0
--     |||  |||||+----- 0
--     |||  ||||+------ 0
--     |||  |||+------- 1
--     |||  ||+-------- 0
--     |||  |+--------- 0
--     |||  +---------- 1
--     |||
--     ||+------------- 0
--     |+-------------- 0
--     +--------------- 0
       13   03  0
  12   20  0
--    1098 76543210   IH_AWIDTH - 320 H = 800 D   -> Hor Addr Time
--     |||  ||||||||
--     |||  |||||||+--- 0
--     |||  ||||||+---- 0
--     |||  |||||+----- 0
--     |||  ||||+------ 0
--     |||  |||+------- 0
--     |||  ||+-------- 0
--     |||  |+--------- 0
--     |||  +---------- 1
--     |||
       ||+------------- 0
       |+-------------- 1
       +--------------- 0
  15   34  0
       14   20  0
--    1098 76543210   IH_TOTAL - 420 H = 1056 D   -> Hor Total Time
--     |||  ||||||||
--     |||  |||||||+--- 0
--     |||  ||||||+---- 0
--     |||  |||||+----- 0
--     |||  ||||+------ 0
--     |||  |||+------- 0
--     |||  ||+-------- 1
--     |||  |+--------- 0
--     |||  +---------- 0
--     |||
--     ||+------------- 1
--     |+-------------- 1
--     +--------------- 0
       17   00  0
  16   05  0
--    1098 76543210   IV_ASTART - 05 H = 5 D   -> ~~Hor Sync Time + H Back Porch~~
--     |||  ||||||||
--     |||  |||||||+--- 1
--     |||  ||||||+---- 1
--     |||  |||||+----- 0
--     |||  ||||+------ 0
--     |||  |||+------- 0
--     |||  ||+-------- 1
--     |||  |+--------- 0
--     |||  +---------- 0
--     |||
       ||+------------- 0
       |+-------------- 0
       +--------------- 0
       19   02  0
  18   58  0
--    1098 76543210   IV_AWIDTH - 258 H = 600 D   -> Ver Addr Time
--     |||  ||||||||
--     |||  |||||||+--- 0
--     |||  ||||||+---- 1
--     |||  |||||+----- 1
--     |||  ||||+------ 1
--     |||  |||+------- 1
--     |||  ||+-------- 0
--     |||  |+--------- 0
--     |||  +---------- 0
--     |||
--     ||+------------- 0
--     |+-------------- 0
--     +--------------- 0
  1B   02  0
       1A   74  0
--    1098 76543210   IV_TOTAL - 274 H = 628 D   -> ~~IV TOTAL~~ Ver Total Time
--     |||  ||||||||
--     |||  |||||||+--- 1
--     |||  ||||||+---- 0
--     |||  |||||+----- 1
--     |||  ||||+------ 1
--     |||  |||+------- 0
```

Disclosed To And Understood By Me

Name	Date

Name	Date

Signature: Blair

Date: 2000/7/28

17168

Subject: ____________________

```
--       1C   8C 0
--     1098 76543210   IH_PULW - 80 H = 128 D  -> HSY Sync Time
--      |||  ||||||||
--      |||  |||||||+--- 0
--      |||  ||||||+---- 0
--      |||  |||||+----- 0
--      |||  ||||+------ 0
--      |||  |||+------- 0
--      |||  ||+-------- 1
--      |||  |+--------- 1
--      |||  +---------- 0
--      |||
--      ||+------------- 0
--      |+-------------- 0
--      +--------------- 0
---------------------------------------------------------------
-- output window parameter
---------------------------------------------------------------
-- PH_ASTART 0128  H = 296 D
  31 01 0
  30 28 0
-- PH_AWIDTH 0400  H = 1024 D
  33 04 0
  32 00 0
-- PV_AWIDTH 0300  H = 768 D
  37 03 0
  36 00 0
-- PV_TOTAL   = 320 H = 800 D
  39 03 0
  38 20 0
-- PH_PULW  = 80 H = 128 D
  3A 80 0
-- PV_PULW  = 06 H = 6 D
  3B 06 0
-- PH_TOTAL  = 542 H = 1346 D  -> Ver Total Time
  C5 05 0
  C4 42 0
-- PH_DELAY  = FF H = 255 D
  C7 00 0
  C6 FF 0
-- PV_BACKPORCH  = 0F H = 15 D
  C3 0F 0
---------------------------------------------------------------
-- background window parameter
---------------------------------------------------------------
-- BH_START 0128 H = 296 D   } 1320 - 296 = 1024
  41 01 0
  40 28 0
-- BH_END 0528 H = 1320 D
  43 05 0
  42 28 0
-- PBV_AOFFSET_EVEN
  51 00 0
  50 00 0
-- PBV_AOFFSET_ODD
  53 00 0
  52 00 0
-- PBV_AWIDTH     panel background vertical active region = 300 H = 768 D
  55 03 0
  54 00 0
---------------------------------------------------------------
-- scaling parameter
---------------------------------------------------------------
-- PH_EMU
  61 03 0      339 H = 825 D = H_TOTAL/VSF = IH_TOTAL (1056)/1.28
  60 39 0
-- IH_EMU
  63 0F 0      fff H = 4095 = H_TOTAL x VSF
  62 FF 0
-- Hdx                                       Hdy/Hdx = 16/HSF - SHhinc
  66 02 0 --> SHdx [5:0] = 02
-- Hdy
  67 01 0 --> SHdy [5:0] = 01
-- Hhinc
  68 0C 0 --> SHhinc [4:0] = 0C = floor (16/HSF) = floor (16/1.28) = 12D = 0CH
-- Vdx                                                                  12.5
  69 02 0 --> SVdx [5:0] = 02
-- Vdy
  6A 01 0 --> SVdy [5:0] = 01
-- Vhinc
  6B 0C 0 --> SVhinc [4:0] = 0C
-- AHdx
  6C 19 0 --> AHdx = 25     AHdy/AHdx = 8 x HSF - AHhinc = 10.24 - 10 = 0.24 = 6/25
-- AHdy
  6D 06 0 --> AHdy = 6
-- AHhinc
  6E 0A 0 --> AHhinc = floor (8 x HSF) = floor (8 x 1.28) = floor (10.24) = 10
-- Vnume
  6F 20 0                     Vnume / Vdeno = VSF = 20H/19H = 32D/25D = 1.28
-- Vdeno
  70 19 0
---------------------------------------------------------------
-- begin of test vector
---------------------------------------------------------------
--@vec
```

Disclosed To And Understood By Me

Name	Date

Name	Date

Signature: 3/sir

Date: 2000/7/28

17169

Subject: _____

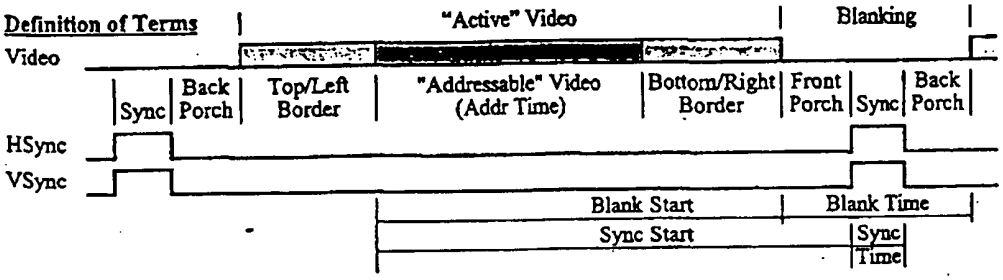
VESA Monitor Timing Specifications Version 1.0 Rev 0.8

VESA MONITOR TIMING STANDARD

Adopted: 8/7/90 (VESA #900602)
Resolution: 800 x 600 at 60 Hz (non-interlaced)
EDID ID: 045h, 040h
BIOS Modes: 102h, 103h, 113h, 114h, & 115h (4, 8, 15, 16, & 24 bpp)

Video Display Information Format (VDIF)
PreAdjusted Timing Data

Timing Name	= 800 x 600 @ 60Hz;					
Hor Pixels	= 800;	// Pixels				
Ver Pixels	= 600;	// Lines				
Hor Frequency	= 37.879;	// KHz	=	26.4 usec	/ line	
Ver Frequency	= 60.317;	// Hz	=	16.6 msec	/ frame	
Pixel Clock	= 40.000;	// MHz	=	25.0 nsec	± 0.5%	
Character Width	= 8;	// Pixels	=	200.0 nsec		
Scan Type	= NONINTERLACED;			// H Phase =	2.3 %	
Hor Sync Polarity	= POSITIVE;	// HBlank	=	24.2% of HTotal		
Ver Sync Polarity	= POSITIVE;	// VBlank	=	4.5% of VTotal		
Hor Total Time	= 26.400;	// (usec)	=	132 chars =	1056 Pixels → IH-TOTAL	
Hor Addr Time	= 20.000;	// (usec)	=	100 chars =	800 Pixels → IH_AWIDTH	
Hor Blank Start	= 20.000;	// (usec)	=	100 chars =	800 Pixels	
Hor Blank Time	= 6.400;	// (usec)	=	32 chars =	256 Pixels	
Hor Sync Start	= 21.000;	// (usec)	=	105 chars =	840 Pixels	
// H Right Border	= 0.000;	// (usec)	=	0 chars =	0 Pixels	
// H Front Porch	= 1.000;	// (usec)	=	5 chars =	40 Pixels → IH_PULW	
Hor Sync Time	= 3.200;	// (usec)	=	16 chars =	128 Pixels → IH_ASTART	
// H Back Porch	= 2.200;	// (usec)	=	11 chars =	88 Pixels	
// H Left Border	= 0.000;	// (usec)	=	0 chars =	0 Pixels	
Ver Total Time	= 16.579;	// (msec)	=	628 lines	HT − (1.06xHA) → IV_TOTAL	
Ver Addr Time	= 15.840;	// (msec)	=	600 lines	= 5.2 → IV_AWIDTH	
Ver Blank Start	= 15.840;	// (msec)	=	600 lines		
Ver Blank Time	= 0.739;	// (msec)	=	28 lines		
Ver Sync Start	= 15.866;	// (msec)	=	601 lines		
// V Bottom Border	= 0.000;	// (msec)	=	0 lines		
// V Front Porch	= 0.026;	// (msec)	=	1 lines		
Ver Sync Time	= 0.106;	// (msec)	=	4 lines		
// V Back Porch	= 0.607;	// (msec)	=	23 lines		
// V Top Border	= 0.000;	// (msec)	=	0 lines		



Disclosed To And Understood By Me

Name _____ Date _____ Signature: _____

Name _____ Date _____ Date: 2003/7/18

17170

Subject: _____

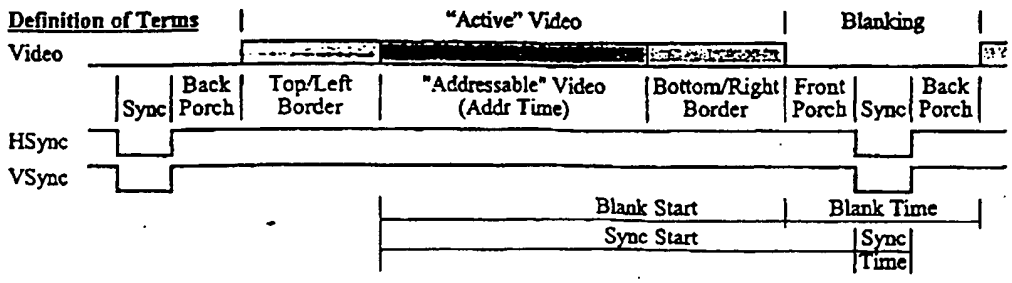
VESA Monitor Timing Specifications Version 1.0 Rev 0.8

VESA MONITOR TIMING STANDARD

Adopted: 9/10/91 (VESA #901101A)
Resolution: 1024 x 768 at 60 Hz (non-interlaced)
EDID ID: 061h, 040h
BIOS Modes: 104h, 105h, 116h, 117h, & 118h (4, 8, 15, 16, & 24 bpp)

Video Display Information Format (VDIF)
PreAdjusted Timing Data

Timing Name	= 1024 x 768 @ 60Hz;						
Hor Pixels	= 1024;	// Pixels					
Ver Pixels	= 768;	// Lines					
Hor Frequency	= 48.363;	// KHz	=	20.7 usec	/	line	
Ver Frequency	= 60.004;	// Hz	=	16.7 msec	/	frame	
Pixel Clock	= 65.000;	// MHz	=	15.4 nsec	±	0.5%	
Character Width	= 8;	// Pixels	=	123.1 nsec			
Scan Type	= NONINTERLACED;		// H Phase	=	5.1 %		
Hor Sync Polarity	= NEGATIVE;	// HBlank	=	23.8% of HTotal			
Ver Sync Polarity	= NEGATIVE;	// VBlank	=	4.7% of VTotal			
Hor Total Time	= 20.677;	// (usec)	=	168 chars	=	1344 Pixels	→ PH_TOTAL
Hor Addr Time	= 15.754;	// (usec)	=	128 chars	=	1024 Pixels	→ PH_ANDTH
Hor Blank Start	= 15.754;	// (usec)	=	128 chars	=	1024 Pixels	
Hor Blank Time	= 4.923;	// (usec)	=	40 chars	=	320 Pixels	
Hor Sync Start	= 16.123;	// (usec)	=	131 chars	=	1048 Pixels	
// H Right Border	= 0.000;	// (usec)	=	0 chars	=	0 Pixels	
// H Front Porch	= 0.369;	// (usec)	=	3 chars	=	24 Pixels	
Hor Sync Time	= 2.092;	// (usec)	=	17 chars	=	136 Pixels	→ PH_PULSW
// H Back Porch	= 2.462;	// (usec)	=	20 chars	=	160 Pixels	→ PH_BSTRT
// H Left Border	= 0.000;	// (usec)	=	0 chars	=	0 Pixels	
Ver Total Time	= 16.666;	// (msec)	=	806 lines		HT – (1.06xHA)	→ PV_TOTAL
Ver Addr Time	= 15.880;	// (msec)	=	768 lines		= 3.98	→ PV_ANDTH
Ver Blank Start	= 15.880;	// (msec)	=	768 lines			
Ver Blank Time	= 0.786;	// (msec)	=	38 lines			
Ver Sync Start	= 15.942;	// (msec)	=	771 lines			
// V Bottom Border	= 0.000;	// (msec)	=	0 lines			
// V Front Porch	= 0.062;	// (msec)	=	3 lines			
Ver Sync Time	= 0.124;	// (msec)	=	6 lines			→ PV_PULSW
// V Back Porch	= 0.600;	// (msec)	=	29 lines			
// V Top Border	= 0.000;	// (msec)	=	0 lines			



Disclosed To And Understood By Me

Name _____ Date _____ Signature: _____

Name _____ Date _____ Date: 2/20/18

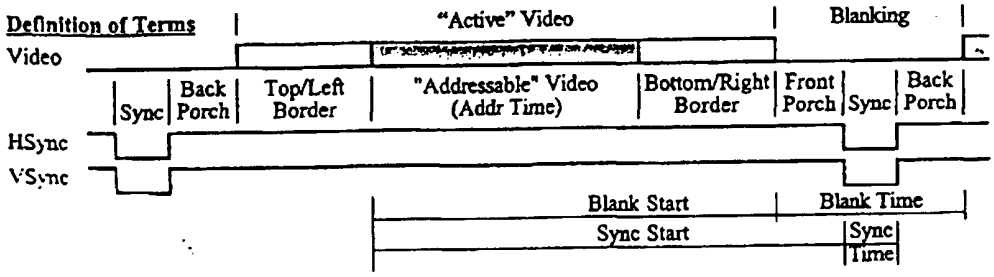
Subject: VGA (640x480) to XGA (1024x768) test.vec parameter setup 17171

HSF = 1024/640 = 1.6 VSF = 768/480 = 1.6

Resolution: 640 x 480 at 60 Hz (non-interlaced)
EDID ID: 031h, 040h
BIOS Modes: 11h, 12h, 101h, 110h, 111h, & 112h (1, 4, 8, 15, 16, & 24 bpp)

Video Display Information Format (VDIF)
PreAdjusted Timing Data

Timing Name	= 640 x 480 @ 60Hz;						
Hor Pixels	= 640;	// Pixels					
Ver Pixels	= 480;	// Lines					
Hor Frequency	= 31.469;	// KHz	=	31.8 usec	/ line		
Ver Frequency	= 59.940;	// Hz	=	16.7 msec	/ frame		
Pixel Clock	= 25.175;	// MHz	=	39.7 nsec	± 0.5%	→ 39.72 ±0.5%	
Character Width	= 8;	// Pixels	=	317.8 nsec			
Scan Type	= NONINTERLACED;			// H Phase =	2.0 %		
Hor Sync Polarity	= NEGATIVE;	// HBlank	=	18.0% of HTotal			
Ver Sync Polarity	= NEGATIVE;	// VBlank	=	5.5% of VTotal			
Hor Total Time	= 31.778;	// (usec)	=	100 chars =	800 Pixels	→ IH_TOTAL	
Hor Addr Time	= 25.422;	// (usec)	=	80 chars =	640 Pixels	→ IH_ADDTW	
Hor Blank Start	= 25.740;	// (usec)	=	81 chars =	648 Pixels		
Hor Blank Time	= 5.720;	// (usec)	=	18 chars =	144 Pixels		
Hor Sync Start	= 26.058;	// (usec)	=	82 chars =	656 Pixels		
// H Right Border	= 0.318;	// (usec)	=	1 chars =	8 Pixels		
// H Front Porch	= 0.318;	// (usec)	=	1 chars =	8 Pixels		
Hor Sync Time	= 3.813;	// (usec)	=	12 chars =	96 Pixels	→ IH_PULW	
// H Back Porch	= 1.589;	// (usec)	=	5 chars =	40 Pixels	136 → IH_STARTT	
// H Left Border	= 0.318;	// (usec)	=	1 chars =	8 Pixels		
Ver Total Time	= 16.683;	// (msec)	=	525 lines	HT – (1.06xHA)	→ IV_TOTAL	
Ver Addr Time	= 15.253;	// (msec)	=	480 lines	= 4.83	→ IV_WIDTH	
Ver Blank Start	= 15.507;	// (msec)	=	488 lines			
Ver Blank Time	= 0.922;	// (msec)	=	29 lines			
Ver Sync Start	= 15.571;	// (msec)	=	490 lines			
// V Bottom Border	= 0.254;	// (msec)	=	8 lines			
// V Front Porch	= 0.064;	// (msec)	=	2 lines			
Ver Sync Time	= 0.064;	// (msec)	=	2 lines			
// V Back Porch	= 0.794;	// (msec)	=	25 lines			
// V Top Border	= 0.254;	// (msec)	=	8 lines			



Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 2003/7/16

17172

Subject:

iclk_prd · IH_TOTAL ÷ VSF = pclk_prd · PH_TOTAL

↑ unknown

$PH\_TOTAL = \frac{iclk\_prd}{pclk\_prd} \cdot IH\_TOTAL \div$ ~~VSF~~ ~~VSF~~ VSF

$= \frac{37.72}{15.33} \times \frac{800}{1.6} = 1295.5 << 1344$

1296

↑ 14.77

因 CRT Cathod, VESA 規定 IH_TOTAL >> IH_WIDTH

但 LCD monitor 則不受此限, 故 IH_TOTAL 可少, 以 Timing 為主

∴ PH_TOTAL = 1296 PH_AWIDTH = 1024 不變

1296 − 1024 = 272 > PH_ASTART

PH_PULW → 128 不變

設 PH_ASTART = 256

PH_DELAY = 257

BH_START = 256

BH_END = 1024 + 256 = 1280

PV_AWIDTH = 300H = 768D

PV_TOTAL = ~~iclk_prd / pclk_prd~~ · IV_TOTAL X VSF = ~~39.72 / 15.33 · 525 / 1.6~~

= ~~850.27~~ 840

PV_PULW = 06

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 200/7/31

Subject:

PV_BACK PORCH = 0F

PH_EMU = IH_TOTAL / VSF = 800/1.6 = 500 = 1F4 (H)

IH_EMU = → 無用, 可不設 (以 PBV_AOFFSET_EVEN/ODD 代替.

~~Hdx~~

$H\,hinc = floor(16/HSF) = 10$

$Hdy/Hdx = \frac{16}{HSF} - H.hinc = 0$

=> Hdx = 1 (任意值)

Hdy = 0

Vhinc = 10

Vdx = 1

Vdy = 0

$AHhinc = floor(8 \times HSF) = floor(8 \times 1.6) = floor(12.8) = 12$

$AHdy/AHdx = 8 \times HSF - AHhinc = 0.8 = \frac{8}{10} = \frac{4}{5}$

=> AHdy = 4

AHdx = 5

$\frac{Vnume}{Vdeno} = VSF = 1.6 = \frac{16}{10} = \frac{8}{5}$

=> Vnume = 8

Vdeno = 5

Disclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2000/7/31

17174

Subject: Brightness Contrast verify

R contrast ⇐ RCAB_CBUS (7~0) [YUV_RGB_adjust_top/ DM_GEN]

R Brightness ⇐ RCAB_CBUS (15~8) G B 开阳

RCAB_CBUS (7~0) ⇐ R CONTRAST_DATA [gc-top/hostif.gc]

RCAB_CBUS (15~8) ⇐ RBRIGHTNESS_DATA

when ADDR_IN = RCONTRAST_ADDR ⇒ R CONTRAST_EN = 1

● RCONTRAST_DATA ⇐ HOST_IBUSN (7~0) [DEC_REG_I-ADDR]

~~DATA_IN~~

HOST_IBUSN (7~0) ⇐ DATA_IN [hostif.gc]

DATA_IN ⇐ HOST_IBUS (15~8)

ADDR_IN ⇐ HOST_IBUS (7~0)

HOST_IBUS (15~8) ⇐ DATA_OUT [I2C (CHIP_CORE)]

● HOST_IBUS (7~1) ⇐ SUB_ADR

CONTRAST :	FF	80	7F	00	
	127/128	0/128	255/128	128/128	← real
		128/128	127/128	0/128	← internal signal
Brightness	-1	-128	127	0	

sclosed To And Understood By Me

Name Date

Name Date

Signature: blair

Date: 2001/8/1

Subject: _____ 17175

test pattern

normal pms => red => grn => blu => dark (bri -> 0 con -> 0

-> light (bri -> 0 con -> 255) -> dark (con -> 1 bri -> 128)

-> light (con -> 1 bri -> 127) -> light red -> light grn

-> light blu -> light red dk = (grn blu) -> grn (bri ->

-> light grn x = -> light blu x = 2

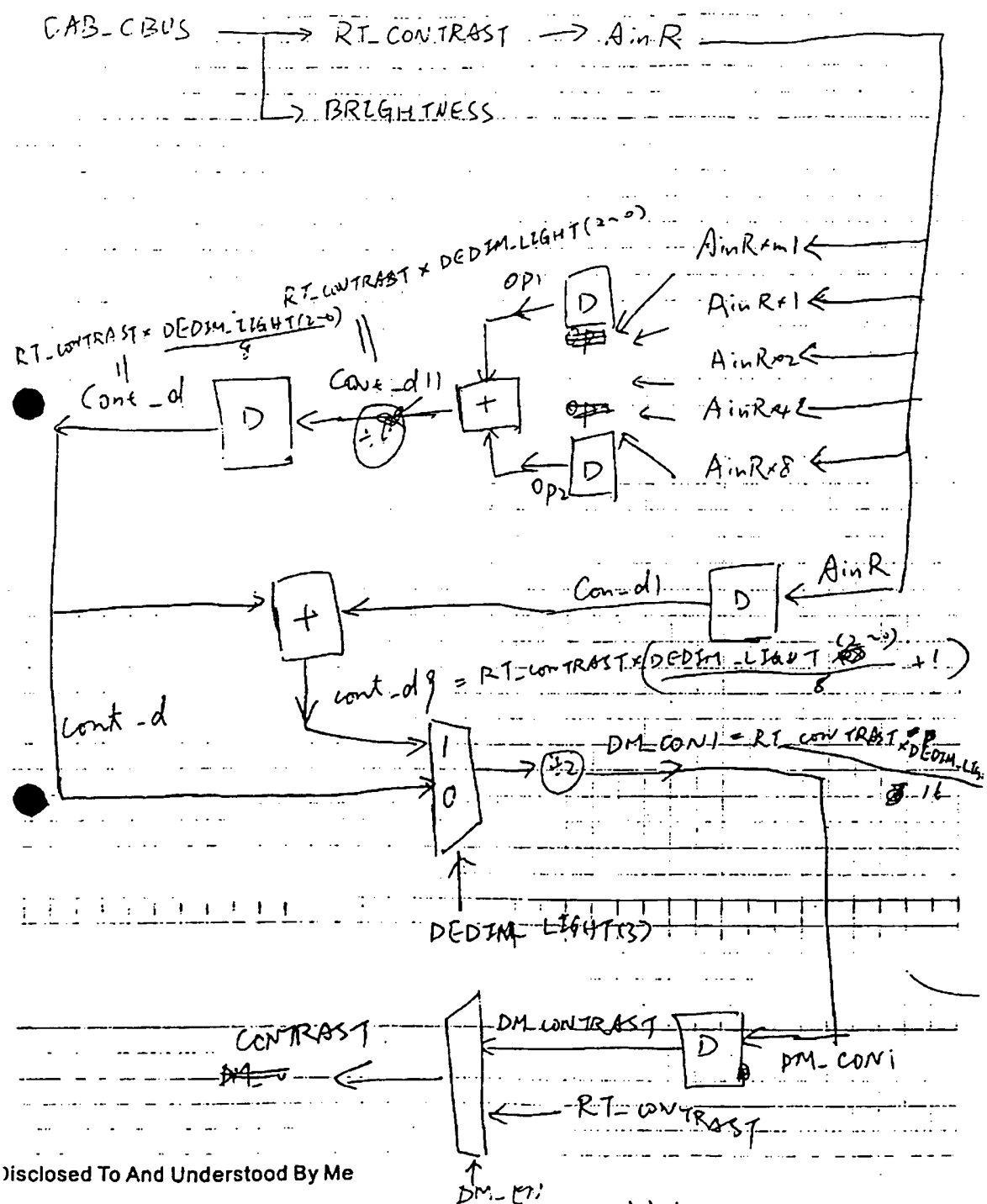
-> normal -> ~~test~~ red overflow

-> grn overflow -> blu overflow

Disclosed To And Understood By Me

Name	Date	Signature: blair
Name	Date	Date: 2001/5/1

Subject: 17176 DM_GEN study



)isclosed To And Understood By Me

Name	Date
Name	Date

Signature: blair

Date: 2000/6/3

Subject: DE_DIM Source code debug

P7177

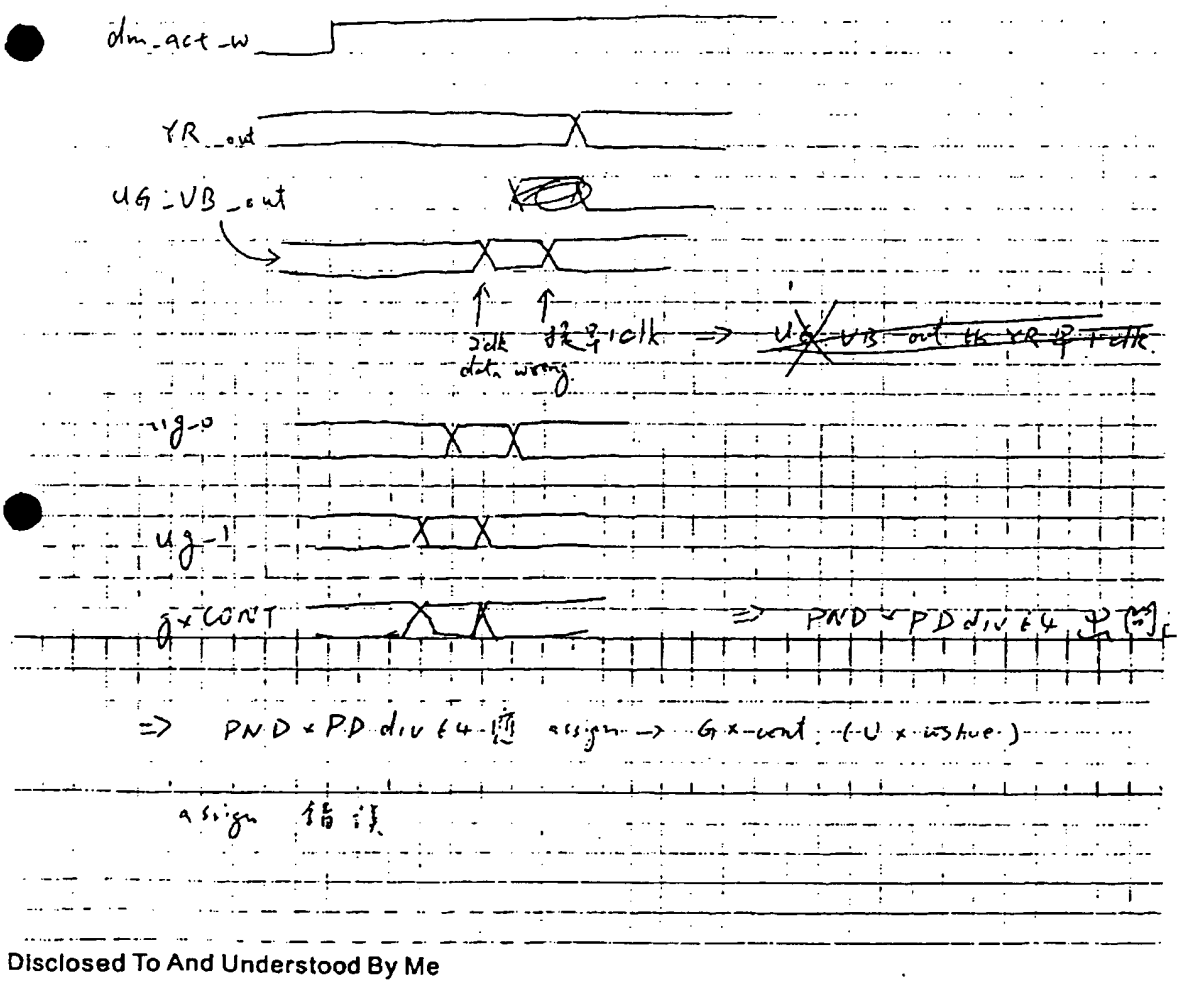
bright_contrast_adjust 中未把 contrast_delay 參數

往下放 boundary_color 偏差

故原因先的 delay 修改 看視 waveform 中

~~R contrast 正~~

YR OK ug. ub wrong



Disclosed To And Understood By Me

Name Date Signature: blair

Name Date Date: 10/1/2

17170

Subject: YUV testing

$Y_{out} = (Y - Y_{black}) \times R_{contrast} + R_{brightness}$

$U_{out} = U \times sat\_coshue + V \times sat\_sinhue$

$V_{out} = V \times sat\_coshue - U \times sat\_sinhue$

Y black → 0 (00), 127 (7F), -128 (80)

Rcontrast → 1 (00), 255/128 (7F), 0 (80)

Rbrightness → 0 (00), 127 (7F), -128 (80)

satcoshue → 1 (40), 0 (00), 127/64 (7F), -2 (80), -1 (C0)

sat-sinhue → 1 (40), 0 (00), 127/64 (7F), -1 (C0), -2 (80)

$Y = 0.299R + 0.587G + 0.114B$

$U = Cb = -0.1687R - 0.3313G + 0.5B$

$V = Cr = 0.5R - 0.4187G - 0.0813B$

isclosed To And Understood By Me

Name Date

Name Date

Signature: 2000/8/6

Date: [illegible]

INTERNATIONAL SEARCH REPORT

ternational Application No
PCT/IB 00/00868

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 C07D277/36 A01N43/78

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 C07D A01N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, CHEM ABS Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 86 07590 A (FMC CORPORATION) 31 December 1986 (1986-12-31) cited in the application the whole document, particularly example 16	1-10
Y	WO 95 24403 A (ZENECA LIMITED) 14 September 1995 (1995-09-14) cited in the application the whole document, particularly pages 15-19, table VII, and examples VII.1-VII.16	1-10

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 September 2000	29/09/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Allard, M

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

ernational Application No
PCT/IB 00/00868

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
WO 8607590 A	31-12-1986	AU	601656 B	13-09-1990
		AU	6122986 A	13-01-1987
		BR	8606746 A	13-10-1987
		CA	1277668 A	11-12-1990
		CN	86104207 A	01-04-1987
		DK	84387 A	19-02-1987
		EP	0228447 A	15-07-1987
		HU	42424 A	28-07-1987
		HU	204022 B	28-11-1991
		OA	8484 A	29-07-1988
		ZA	8604637 A	24-02-1988
		JP	63500037 T	07-01-1988
		MX	9249 A	01-12-1993
		US	4952580 A	28-08-1990
WO 9524403 A	14-09-1995	AP	649 A	29-05-1998
		AU	685242 B	15-01-1998
		AU	1816495 A	25-09-1995
		BG	100900 A	31-07-1997
		BR	9507042 A	09-09-1997
		CA	2182520 A	14-09-1995
		CN	1143958 A	26-02-1997
		CZ	9602632 A	11-12-1996
		EG	20790 A	29-03-2000
		EP	0749433 A	27-12-1996
		FI	963539 A	09-09-1996
		HU	74902 A,B	28-02-1997
		JP	9510197 T	14-10-1997
		LV	11686 A,B	20-02-1997
		NO	963776 A	07-11-1996
		NZ	281267 A	24-11-1997
		PL	316175 A	23-12-1996
		SK	114896 A	09-04-1997
		US	5705516 A	06-01-1998
		US	5912243 A	15-06-1999

SEARCH REPORT

Information on patent family members

Application No. 200100382-1

This annex lists the patent family members relating to the patent documents cited in the search report.
The members are as contained in the EPIDOS INPADOC file.
The Office is in no way liable for these particulars which are merely given for the purpose of information

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO A1 9422825	13-10-1994	AT E 188692	15-01-2000
		AU A1 66218/94	24-10-1994
		AU B2 683882	27-11-1997
		CA AA 2159482	13-10-1994
		DE C0 69422615	17-02-2000
		DE T2 69422615	03-08-2000
		EP A4 693067	12-12-1995
		EP A1 693067	24-01-1996
		EP B1 693067	12-01-2000
		JP T2 8508501	10-09-1996
		US A 5441952	15-08-1995
WO A1 9515963	15-06-1995	DE A1 4341665	08-06-1995
		DE C0 59409541	02-11-2000
		EP A1 733052	25-09-1996
		EP B1 733052	27-09-2000
		JP T2 9506102	17-06-1997
		TW B 383298	01-03-2000
		US A 5693637	02-12-1997
WO A2 9300095	07-01-1993	AU A1 22711/92	25-01-1993
WO A3 9300095	18-02-1993	AU B2 666318	08-02-1996
		EP A1 593603	27-04-1994
		EP B1 593603	20-11-2002
		IE A1 922087	30-12-1992
		JP T2 6509074	13-10-1994
		NZ A 243326	26-10-1995
		PT A 100631	30-09-1993
		PT B 100631	30-06-1999
		ZA A 9204760	31-03-1993
		US A 5693636	02-12-1997
		US A 6127359	03-10-2000
		US A 5939412	17-08-1999
WO A1 9308174	29-04-1993	AT E 133412	15-02-1996
WO A1 9308174	29-04-1993	AU A1 27976/92	21-05-1993
		AU B2 673501	14-11-1996
		DE C0 69207916	07-03-1996
		DE T2 69207916	17-10-1996
		DK T3 610334	03-06-1996
		EP A1 610334	17-08-1994
		EP B1 610334	24-01-1996
		ES T3 2085651	01-06-1996
		FI A 941222	02-06-1994
		GR T3 3019517	31-07-1996
		JP T2 7500341	12-01-1995
		NO A0 941378	15-04-1994
		NO A 941378	20-06-1994
		NO B1 307417	03-04-2000
		US A 5250679	05-10-1993
		US A 5403836	04-04-1995
		US A 5565449	15-10-1996
		US A 5663166	02-09-1997
		US A 5674863	07-10-1997
		US A 5674865	07-10-1997
		FI A0 941222	15-03-1994
EP A1 635492	25-01-1995	AT E 225337	15-10-2002
EP B1 635492	02-10-2002	AU A1 67500/94	02-02-1995
EP A1 635492	25-01-1995	AU B2 685807	29-01-1998
EP B1 635492	02-10-2002	BR A 9402916	11-04-1995
		CA AA 2128348	23-01-1995
		CN A 1108248	13-09-1995
		CN B 1057292	11-10-2000
		CN A 1274723	29-11-2000
		CZ A3 9401740	13-09-1995
		DE C0 69431462	07-11-2002
		FI A0 943478	22-07-1994
		FI A 943478	23-01-1995
		FI A 20000648	20-03-2000

SEARCH REPORT

Information on patent family members

Application No. 200100382-1

This annex lists the patent family members relating to the patent documents cited in the search report.
The members are as contained in the EPIDOS INPADOC file.
The Office is in no way liable for these particulars which are merely given for the purpose of information

Patent document cited in search report	Publication date	Patent family member(s)			Publication date
		FI	B1	106854	30-04-2001
		FI	B1	106858	30-04-2001
		HU	A0	9402156	28-09-1994
		HU	A2	70397	30-10-1995
		IL	A0	110172	07-10-1994
		IL	A1	110172	31-10-2001
		JP	A2	8188564	23-07-1996
		NO	A0	942734	21-07-1994
		NO	A	942734	23-01-1995
		NO	B1	302885	04-05-1998
		NZ	A	264060	22-08-1997
		PL	A1	304388	23-01-1995
		PL	B1	181905	31-10-2001
		TW	B	450953	21-08-2001
		US	A	5618843	08-04-1997
		US	BA	6472405	29-10-2002
		US	A	5731324	24-03-1998
		US	A	6020362	01-02-2000
		US	A	6137002	24-10-2000
		US	BA	6448269	10-09-2002
		RU	C1	2140907	10-11-1999
		ZA	A	9405251	18-01-1996
		AT	E	220903	15-08-2002
		AU	A1	47580/96	07-08-1996
		AU	B2	706278	10-06-1999
		BR	A	9607570	08-09-1999
		CA	AA	2210682	25-07-1996
		DE	C0	69622532	29-08-2002
		DK	T3	804431	28-10-2002
		EP	A1	804431	05-11-1997
		EP	A4	804431	29-04-1998
		EP	B1	804431	24-07-2002
		FI	A0	972951	11-07-1997
		FI	A	972951	21-08-1997
		HU	AB	9801433	28-05-1999
		IL	A0	116761	14-05-1996
		IL	A0	128421	31-01-2000
		JP	T2	11502194	23-02-1999
		NO	A0	973304	17-07-1997
		NO	A	973304	10-09-1997
		NZ	A	302013	28-01-2000
		RU	C2	2169146	20-06-2001
		TW	B	419466	21-01-2001
		WO	A1	9622288	25-07-1996

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION OF ELECTION

(PCT Rule 61.2)

To:

Commissioner
US Department of Commerce
United States Patent and Trademark
Office, PCT
2011 South Clark Place Room
CP2/5C24
Arlington, VA 22202
ETATS-UNIS D'AMERIQUE

in its capacity as elected Office

Date of mailing: 11 January 2001 (11.01.01)	
International application No.: PCT/IB00/00868	Applicant's or agent's file reference: NIT 364-WO
International filing date: 28 June 2000 (28.06.00)	Priority date: 06 July 1999 (06.07.99)
Applicant: WATANABE, Yukiyoshi et al	

1. The designated Office is hereby notified of its election made:

[X] in the demand filed with the International preliminary Examining Authority on:

01 December 2000 (01.12.00)

[ ] in a notice effecting later election filed with the International Bureau on:

2. The election [X] was

[ ] was not

made before the expiration of 19 months from the priority date or, where Rule 32 applies, within the time limit under Rule 32.2(b).

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No.: (41-22) 740.14.35

Authorized officer:

J. Zahra

Telephone No.: (41-22) 338.83.38

10/030361

SEARCH REPORT

Application No.
0100382-1

A. CLASSIFICATION OF SUBJECT MATTER

According to the International Patent Classification (IPC[7]):

A61K31/5513

B. FIELDS SEARCHED IPC[7]:

A61K

Electronic data base consulted during the search (name of data base and, where practicable, search terms used)
WPI, CAS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 94/22825 A1 (MERCK & CO., INC.) 13 October 1994 (13.10.94) *claim 1.*	2,8,13, 28,29,33
X	WO 95/15963 A1 (BASF AKTIENGESELLSCHAFT) 15 June 1995 (15.06.95) *claim 1.*	2,8-10,13,28, 29,33-35
X	WO 93/00095 A2 (SMITHKLINE BEECHAM CORPORATION) 7 January 1993 (07.01.93) *claims.*	2,8-10,13,28, 29,33-35
X	WO 93/08174 A1 (GENENTECH, INC.) 29 April 1993 (29.04.93) *claim 1.*	2,8-10,13,28, 29,33-35

☒ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the filing date but later than the priority date claimed

"T" later document published after the filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of actual completion of the search: **16 September 2002 (16.09.2002)**

AUSTRIAN PATENT OFFICE
Kohlmarkt 8-10, A-1014 VIENNA
Facsimile No. ++431/53424/535

Authorized Officer KRENN M.

Telephone No. ++431/53424/ **435**

SEARCH REPORT

Application No.
0100382-1

C (Continuation - Page 2). DOCUMENTS CONSIDERED TO BE RELEVANT

X	EP 0635492 A1 (ELI LILLY AND COMPANY) 25 January 1995 (25.01.95) *claims.* ----	2,8,9,13,28, 29,33,34

Austrian Patent Office

Application No. 0100382-1	Applicant BIOGEN, INC.
Filing date 24 July 1997 (24.07.1997)	(Earliest) Priority Date 25 July 1996 (25.07.1996)

SEARCH REPORT

EXPLANATIONS

WO 94/22825 A1, WO 9515963 A1 and WO 9308174 A1 refer to benzodiazepine-2,5-diones useful for inhibiting the binding of fibrinogen to blood platelets and thereby inhibiting the aggregation of blood platelets.

WO 93/00095 A2 describes bicyclic (6 + 7 membered ring) fibrinogen antagonists, wherein the 7-membered ring might contain up to two heteroatoms chosen from the group of O, S and N.

EP 0635492 A1 concerns bicyclic compounds composed of two fused six membered rings useful as glycoprotein IIb/IIIa antagonists.

☐ EXAMINATION REPORT
☒ WRITTEN OPINION

Application No.
0100382-1

Reasoned statement with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

Date of actual completion of the report / opinion: **16 September 2002 (16.09.2002)**

1. STATEMENT

Novelty (N)	YES	Claims	36,46-49
	NO	Claims	2,8-10,13,28,29,33-35
Inventive step (IS)	YES	Claims	36,46-49
	NO	Claims	2,8-10,13,28,29,33-35
Industrial applicability (IA)	YES	Claims	2,8-10,13
	NO	Claims	28,29,33-36,46-49

2. CITATIONS AND EXPLANATIONS

Both WO 9422825 A1 and WO 9515963 A1 refer to variously substituted benzodiazepine-2,5-diones.
Provided that in compound IIa of the present application $A_1 = NR^1$ (R^1 = H); X = O; $R^3 = R^1$ and R^1 = aromatic or non-aromatic ring system; $(CR^1R^2)_m = C_1$–C_4-alkyl and W = H; R^5 = H; R^6 = H and "A" = carboxyalkyl substituted aralkyl, claims 2,8,13,28,29,33 are anticipated by WO 9422825 A1.
Provided that in compound IIa of the present application $A_1 = NR^1$; X = O; $R^3 = R^1$; $(CR^1R^2)_m$ = C_1–C_4-alkyl; A,B,D and E = methylene and "A" = acyl, aroyl, etc., claims 2, 8-10,13,28,29,33-35 are anticipated by WO 9515963 A1.

WO 9300095 A2 refers to bicyclic fibrinogen antagonists, wherein D^1-D^4 = -CH_2-; R^6 = (N-substituted) aminocarbonyl-R; A^1-A^5 form variously substituted seven-membered rings containing up to two heteroatoms chosen from the group O, S and N.

WO 9308174 A1 concerns integrin inhibitors, wherein ring "A" = substituted benzol (Q^1-L^1 = aminocarbonyl); T-U-G = -NCOC- and D = hydrogen.

EP 0635492 A1 discloses glycoprotein IIb/IIIa antagonists showing a bicyclic (= two fused six membered rings, A and B) structure; wherein Q = organic group comprising a basic radical, L = linker (e.g. alkylene) or a bond; R_3 = carboxymethyl and R_0 = =O.

In respect of the above cited literature certain compounds disclosed in claims 2, 8-10,13,28,29,33-35 are neither new nor inventive.

Claim 36 describes compounds, wherein A is selected from the group consisting of (N-Ar'-urea)-para-substituted aryl, aralkyl or aralkylcarbonyl. Said compounds are not state of the art; thus claim 36 involves both novelty and inventive step.

As the compounds disclosed in the cited literature are used either as fibrinogen receptor antagonists or as endothelin receptor antagonist and not for treating asthma, MS, diabetes, inflammatory or autoimmune diseases, claims 46-49 show novelty as well as inventive step.

Claims 28,29,33-36 and 46-49 refer to a method of treatment of the human or animal body by therapy, which is not taken to be capable of industrial application (see Singapore Patents Act, Section 16(2)); thus only claims 2,8-10 and 13 are industrially applicable.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 January 2001 (11.01.2001)

PCT

(10) International Publication Number
WO 01/02378 A1

(51) International Patent Classification[7]: C07D 277/36, A01N 43/78

(21) International Application Number: PCT/IB00/00868

(22) International Filing Date: 28 June 2000 (28.06.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/191638 6 July 1999 (06.07.1999) JP

(71) Applicant *(for all designated States except US)*: **NIHON BAYER AGROCHEM K.K.** [JP/JP]; 10-8, Takanawa 4-chome, Minato-ku, Tokyo 108 (JP).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **WATANABE, Yukiyoshi** [JP/JP]; 2-8-24, Hanagaki-cho, Oyama-shi, Tochigi 323 (JP). **ISHIKAWA, Koichi** [JP/JP]; 5-5-1, Ekinan-cho, Oyama-shi, Tochigi 323-0822 (JP). **OTSU, Yuichi** [JP/JP]; 1-5-7, Ekinan-cho, Oyama-shi, Tochigi 323-0822 (JP). **SHIBUYA, Katsuhiko** [JP/JP]; 6-14-4, Midori, Minamikawachi-machi, Kawachi-gun, Tochigi 329-0433 (JP). **ABE, Takahisa** [JP/JP]; 4-1-24, Miyanomori-2 jo, Chuo-ku, Sapporo-shi, Hokkaido 064-0952 (JP).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

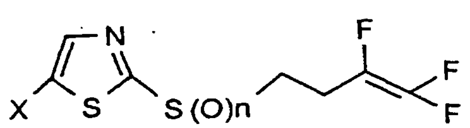
(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 01/02378 A1

(54) Title: NEMATICIDAL TRIFLUOROBUTENES



(I)

(57) **Abstract:** The invention relates to compounds of formula (I) in which X represents halogen, and n represents 0, 1 or 2, to a process for their preparation and to their use as nematicides.

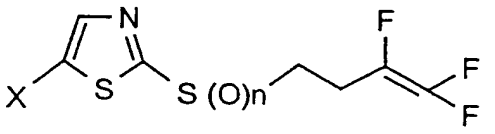
PCT/IB00/00868

NEMATICIDAL TRIFLUOROBUTENES

The present invention relates to novel triflluorobutenes and their use as nematicides.

US Patent No. 3,518,172 describes trifluorobutenyl compounds which have nematicidal activity. Japanese Laid-open Patent Publication (PCT) No. 500037/1988 (= WO 86/07590) also describes that some kinds of polyhaloalkene compounds have nematicidal activity. Further, WO 95/24403 describes that 4,4-difluorobutenyl compounds have nematicidal activity. Japanese Laid-open Patent Application No. 176141/1997 mentiones thiazole derivatives having insecticidal and acaricidal activity.

There have now been found novel trifluorobutenes of the formula (I)



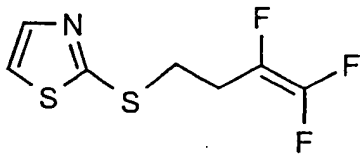
(I)

in which

X represents halogen and

n represents 0, 1 or 2.

The compounds of the formula (I) in which n represents 0 can be obtained when trifluorobutenes of the formula (Ia)



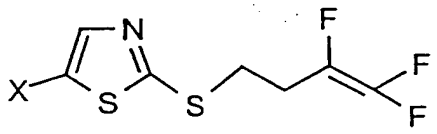
(Ia)

are reacted with a halogenating agent, optionally in the presence of one or more inert diluents (process (A)).

The compounds of the formula (I) in which

n represents 1 or 2

can be obtained when compounds of the formula (Ib)



(Ib)

in which

X is the same as defined above

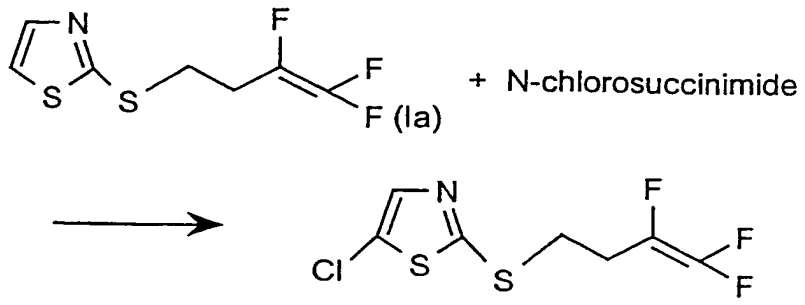
are reacted with an oxidizing agent, optionally in the presence of one or more inert diluents (process (B)).

The compounds of the formula (I) of the present invention have strong nematicidal activity and show good compatibility with various crops. According to the present invention the compounds of the formula (I) have surprisingly strong nematicidal activity compared with the known compounds described in the aforementioned literature.

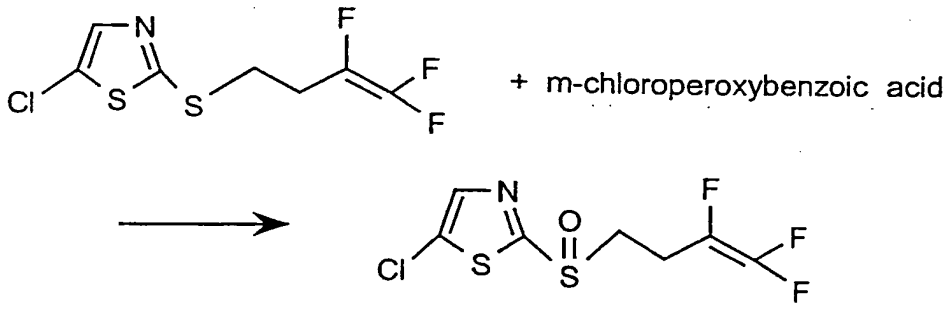
In the present specification X preferably represents fluoro, chloro or bromo. X particularly preferably represents fluoro or chloro. X very particularly preferably represents chloro.

In the present specification n preferably represents 0 or 2. n particulary preferably represents 2.

Process (A) for preparing compounds of the formula (I) of the present invention can be represented by the following reaction scheme in which N-chlorosuccinimide is exemplaryly used as halogenating agent:



Process (B) for preparing compounds of the formula (I) of the present invention can be represented by the following reaction in which 5-chloro-2-(3,4,4-trifluoro-3-butenylthio)thiazole is used as a starting material and m-chloroperoxybenzoic acid is exemplaryly used as oxidizing agent.



2-(3,4,4-trifluoro-3-butenylthio)-thiazole is a known compound described in Japanese Laid-open Patent Publication (PCT) No. 500037/1988 (= WO 86/07590). Compounds of formula (Ib), which are used as starting material in process (B), correspond to the compounds of the formula (I) of the present invention in which n represents 0 and can be synthesized according to the aforementioned process (A).

Halogenating agents used in Process (A) can be agents usually used for this purpose in organic chemistry and which are known to a person skilled in the art, including for example sulfuryl chloride, N-chlorosuccinimide, N-bromosuccinimide, trichloro-isocyanuric acid, potassium fluoride, sodium chlorate, phosphorus pentachloride, titanium (IV) chloride, chlorine gas, bromine, iodine etc.

Oxidizing agents used for the oxidation of the above-mentioned compounds of the formula (Ib) in process (B) can be agents usually used for this purpose in organic chemistry and which are known to a person skilled in the art including for example hydrogen peroxide water, m-chloroperoxybenzoic acid, peroxyacetic acid, peroxy-benzoic acid, magnesium monoperoxyphthalate, potassium peroxymonosulfate, etc.

The reaction of the above-mentioned process (A) is preferably conducted in the presence of an adequate diluent. Diluents which can be used in this process can for example be water; aliphatic, alicyclic and aromatic hydrocarbons (which can be optionally chlorinated) such as hexane, cyclohexane, petroleum ether, ligroine, benzene, methylene chloride, chloroform, carbon tetrachloride, ethylene chloride, chlorobenzene etc.; ethers, such as diethyl ether, methyl ethyl ether, di-isopropyl ether, dibutyl ether, propylene oxide, dioxane, tetrahydrofuran etc.; nitriles, such as acetonitrile, propionitrile, acrylonitrile etc.; acid amides, such as dimethylformamide, dimethylacetamide, N-methylpyrrolidone etc.; sulfones and sulfoxides, such as dimethyl sulfoxide, sulfolane etc.

The reaction temperatures of process (A) according to the invention can be varied over a relatively wide range. In general, temperatures in a range of between 0 and 200°C, preferably between 20 and 150°C are employed. The process (A) according to the invention is generally carried out under normal pressure. However, it is possible to carry out the process (A) under elevated pressure or under reduced pressure, in general between 0.1 bar and 10 bar.

To carry out the process (A) according to the invention, the starting materials are

generally employed in approximately equimolar amounts. However, it is also possible to use one of the components in a relatively large excess. Work-up is carried out according to customary methods (cf. the preparation examples).

For example, the compound of the formula (I) in which n represents 0 and X represents chloro can be obtained by reacting 1-1.2 moles of N-chlorosuccinimide with 1 mole of 2-(3,4,4-trifluoro-3-butenylthio)thiazole in carbon tetrachloride under reflux by heating.

The reaction of the above-mentioned process (B) is preferably conducted in the presence of an adequate diluent. Diluents which can be used in this process can for example be water; aliphatic, alicyclic and aromatic hydrocarbons (which can be optionally chlorinated), such as hexane, cyclohexane, petroleum, ether, ligroine, benzene, toluene, xylene, methylene chloride, chloroform, carbon tetrachloride, ethylene chloride, chlorobenzene etc.; ethers, such as diethyl ether, methyl ethyl ether, di-isopropyl ether, dibutyl ether, propylene oxide, dioxane, tetrahydrofuran etc.; nitriles, such as acetonitrile, propionitrile, acrylonitrile etc.; alcohols, for example methanol, ethanol, isopropanol, butanol, ethylene glycol etc.; esters, for example ethyl acetate, amyl acetate etc.; acid amides, for example dimethyl-formamide, dimethylacetamide, N-methylpyrrolidone etc.; sulfones and sulfoxides, for example dimethyl sulfoxide, sulfolane etc.; carboxylic acids, for example formic acid, acetic acid etc.

The reaction temperatures of process (B) according to the invention can be varied over a relatively wide range. In general, temperatures in a range of between 0 and 150°C, preferably between 0 and 120°C are employed. The process (B) according to the invention is generally carried out under normal pressure. However, it is also possible to carry out the process (B) under elevated pressure or under reduced pressure, in general between 0.1 bar and 10 bar.

To carry out the process (B) according to the invention, the starting materials are generally employed in approximately equimolar amounts. However, it is also

possible to use one of the components in a relatively large excess. Work-up is carried out according to customary methods (cf. the preparation examples).

For example, compounds of the formula (I) in which n represents 1 can be obtained by reacting, 1-2 moles of m-chloroperoxybenzoic acid with 1 mole of the compound of the formula (Ib) in methylene chloride under cooling with ice.

The compounds of the formula (I) according to the present invention show strong controlling activity against nematodes. They can, therefore, be efficiently used as nematicidal agents. The compounds of the formula (I) of the present invention do not exhibit phytotoxicity against crops and can be used for controlling harmful nematodes.

The compounds according to the invention can be used, for example, against nematodes such as Pratylenchus spp., Globodera spp., such as Globodera rostochiensis wollenweber, Heterodera spp., such as Heterodera glycines ichinohe, Meloidogyne spp., Aphelenchoides spp., such as Aphelenchoides basseyi christie, Radopholus similis, Ditylenchus dipsaci, Tylenchulus semipenetrans, Longidorus spp., Xiphinema spp., Trichodorus spp., Bursaphelenchus spp., such as Bursaphelenchus xylophilis etc.

The compounds according to the invention are especially useful for combating Pratylenchus spp., Globodera rostochiensis wollenweber, Heterodera glycines ichinohe, Meloidogyne spp., Aphelenchoides basseyi christie, Bursaphelenchus xylophilis.

However, the use of the active compounds according to the invention is in no way restricted to these genera, but also extends in the same manner to other nematodes.

The active compounds can be converted into the customary formulations, such as solutions, emulsions, wettable powders, water dispersible granules, suspensions,

powders, dusting agents, foaming agents, pastes, soluble powders, granules, suspo-emulsion concentrates, microcapsules, fumigants, natural and synthetic materials impregnated with active compound and very fine capsules and polymeric substances.

These formulations are prepared in a known manner, for example by mixing the active compounds with extenders, that is liquid solvents, liquefied gas and/or solid diluents or carriers, if appropriate with the use of surface-active agents, that is emulsifiers and/or dispersants and/or foam-formers.

If the extender used is water, it is also possible to use, for example, organic solvents as auxiliary solvents. Suitable liquid solvents are essentially: aromatics, such as xylene, toluene, or alkylnaphthalenes, chlorinated aromatics and chlorinated aliphatic hydrocarbons, such as chlorobenzene, chloroethylenes or methylene chloride, aliphatic hydrocarbons, such as cyclohexane or paraffins, for example mineral oil fractions, mineral or vegetable oil, alcohols, such as butanol or glycol, and also their ethers and esters, ketones, such as acetone, methyl ethyl ketone, methyl isobutyl ketone or cyclohexanone, strongly polar solvents, such as dimethylformamide and dimethyl sulphoxide, and also water.

Liquefied gas diluents or carriers are liquefied substances which are gases at normal temperature and pressure. Liquefied gas diluents can be, for example, aerosol propellants such as butane, propane, nitrogen gas, carbon dioxide, halogenated hydrocarbons, etc.

Suitable solid carriers are:

for example ammonium salts and ground natural minerals, such as kaolins, clays, talc, chalk, quartz, attapulgite, montmorillonite or diatomaceous earth, and ground synthetic minerals, such as finely divided silica, alumina and silicates; suitable solid carriers for granules are: for example crushed and fractionated natural rocks such as calcite, marble, pumice, sepiolite and dolomite, as well as synthetic granules of inorganic and organic meals, and granules of organic material such as sawdust,

coconut shells, maize cobs and tobacco stalks; suitable emulsifiers and/or foam-formers are: for example nonionic and anionic emulsifiers, such as polyoxyethylene fatty acid esters, polyoxyethylene fatty alcohol ethers, for example alkylaryl polyglycol ethers, alkylsulphonates, alkyl sulphates, arylsulphonates and protein hydrolysates; suitable dispersants are: for example lignin-sulphite waste liquors and methylcellulose.

Tackifiers such as carboxymethylcellulose and natural and synthetic polymers in the form of powders, granules or latices, such as gum arabic, polyvinyl alcohol and polyvinyl acetate, as well as natural phospholipids, such as cephalins and lecithins, and synthetic phospholipids, can be used in the formulations. Other additives can be mineral and vegetable oils.

It is possible to use colorants such as inorganic pigments, for example iron oxide, titanium oxide and Prussian Blue, and organic dyestuffs, such as alizarin dyestuffs, azo dyestuffs and metal phthalocyanine dyestuffs, and trace nutrients such as salts of iron, manganese, boron, copper, cobalt, molybdenum and zinc.

The formulations in general contain between 0.01 and 95 per cent by weight of active compound, preferably between 0.1 and 90%, particularly preferably between 0.5 and 90%.

The active compounds according to the invention, as such or in their formulations, can also be used in a mixture with known fungicides, bactericides, acaricides, nematicides or insecticides, to widen, for example, the activity spectrum or to prevent the development of resistance. In many cases, this results in synergistic effects, i.e. the activity of the mixture exceeds the activity of the individual components.

Examples of particularly advantageous mixing components are the following:

Fungicides:

aldimorph, ampropylfos, ampropylfos potassium, andoprim, anilazine, azaconazole, azoxystrobin,

benalaxyl, benodanil, benomyl, benzamacril, benzamacril-isobutyl, bialaphos, binapacryl, biphenyl, bitertanol, blasticidin-S, bromuconazole, bupirimate, buthiobate,

calcium polysulphide, capsimycin, captafol, captan, carbendazim, carboxin, carvon, quinomethionate, chlobenthiazone, chlorfenazole, chloroneb, chloropicrin, chlorothalonil, chlozolinate, clozylacon, cufraneb, cymoxanil, cyproconazole, cyprodinil, cyprofuram,

debacarb, dichlorophen, diclobutrazole, diclofluanid, diclomezine, dicloran, diethofencarb, difenoconazole, dimethirimol, dimethomorph, diniconazole, diniconazole-M, dinocap, diphenylamine, dipyrithione, ditalimfos, dithianon, dodemorph, dodine, drazoxolon,

ediphenphos, epoxiconazole, etaconazole, ethirimol, etridiazole,

famoxadon, fenapanil, fenarimol, fenbuconazole, fenfuram, fenitropan, fenpiclonil, fenpropidin, fenpropimorph, fentin acetate, fentin hydroxide, ferbam, ferimzone, fluazinam, flumetover, fluoromide, fluquinconazole, flurprimidol, flusilazole, flusulfamide, flutolanil, flutriafol, folpet, fosetyl-aluminium, fosetyl-sodium, fthalide, fuberidazole, furalaxyl, furametpyr, furcarbonil, furconazole, furconazole-cis, furmecyclox,

guazatine,

hexachlorobenzene, hexaconazole, hymexazole,

imazalil, imibenconazole, iminoctadine, iminoctadine albesilate, iminoctadine triacetate, iodocarb, ipconazole, iprobenfos (IBP), iprodione, irumamycin, isoprothiolane, isovaledione,

kasugamycin, kresoxim-methyl, copper preparations, such as: copper hydroxide, copper naphthenate, copper oxychloride, copper sulphate, copper oxide, oxine-copper and Bordeaux mixture,

mancopper, mancozeb, maneb, meferimzone, mepanipyrim, mepronil, metalaxyl, metconazole, methasulfocarb, methfuroxam, metiram, metomeclam, metsulfovax, mildiomycin, myclobutanil, myclozolin,

nickel dimethyldithiocarbamate, nitrothal-isopropyl, nuarimol,

ofurace, oxadixyl, oxamocarb, oxolinic acid, oxycarboxim, oxyfenthiin,

paclobutrazole, pefurazoate, penconazole, pencycuron, phosdiphen, pimaricin, piperalin, polyoxin, polyoxorim, probenazole, prochloraz, procymidone, propamocarb, propanosine-sodium, propiconazole, propineb, pyrazophos, pyrifenox, pyrimethanil, pyroquilon, pyroxyfur,

quinconazole, quintozene (PCNB),

sulphur and sulphur preparations,

tebuconazole, tecloftalam, tecnazene, tetcyclacis, tetraconazole, thiabendazole, thicyofen, thifluzamide, thiophanate-methyl, thiram, tioxymid, tolclofos-methyl, tolylfluanid, triadimefon, triadimenol, triazbutil, triazoxide, trichlamide, tricyclazole, tridemorph, triflumizole, triforine, triticonazole,

uniconazole,

validamycin A, vinclozolin, viniconazole,

zarilamide, zineb, ziram and also

Dagger G,

OK-8705,

OK-8801,

α-(1,1-dimethylethyl)-∃-(2-phenoxyethyl)-1H-1,2,4-triazole-1-ethanol,

α-(2,4-dichlorophenyl)-∃-fluoro-b-propyl-1H-1,2,4-triazole-1-ethanol,

α-(2,4-dichlorophenyl)-∃-methoxy-a-methyl-1H-1,2,4-triazole-1-ethanol,

α-(5-methyl-1,3-dioxan-5-yl)-∃-[[4-(trifluoromethyl)-phenyl]-methylene]-1H-1,2,4-triazole-1-ethanol,

(5RS,6RS)-6-hydroxy-2,2,7,7-tetramethyl-5-(1H-1,2,4-triazol-1-yl)-3-octanone,

(E)-a-(methoxyimino)-N-methyl-2-phenoxy-phenylacetamide,

isopropyl 1-{2-methyl-1-[[[1-(4-methylphenyl)-ethyl]-amino]-carbonyl]-propyl}-carbamate,

1-(2,4-dichlorophenyl)-2-(1H-1,2,4-triazol-1-yl)-ethanone O-(phenylmethyl) oxime,

1-(2-methyl-1-naphthalenyl)-1H-pyrrol-2,5-dione,

1-(3,5-dichlorophenyl)-3-(2-propenyl)-2,5-pyrrolidinedione,
1-[(diiodomethyl)-sulphonyl]-4-methyl-benzene,
1-[[2-(2,4-dichlorophenyl)-1,3-dioxolan-2-yl]-methyl]-1H-imidazole,
1-[[2-(4-chlorophenyl)-3-phenyloxiranyl]-methyl]-1H-1,2,4-triazole,
1-[1-[2-[(2,4-dichlorophenyl)-methoxy]-phenyl]-ethenyl]-1H-imidazole,
1-methyl-5-nonyl-2-(phenylmethyl)-3-pyrrolidinole,
2',6'-dibromo-2-methyl-4'-trifluoromethoxy-4'-trifluoro-methyl-1,3-thiazole-5-carboxanilide,
2,2-dichloro-N-[1-(4-chlorophenyl)-ethyl]-1-ethyl-3-methyl-cyclopropane-carboxamide,
2,6-dichloro-5-(methylthio)-4-pyrimidinyl thiocyanate,
2,6-dichloro-N-(4-trifluoromethylbenzyl)-benzamide,
2,6-dichloro-N-[[4-(trifluoromethyl)-phenyl]-methyl]-benzamide,
2-(2,3,3-triiodo-2-propenyl)-2H-tetrazole,
2-[(1-methylethyl)-sulphonyl]-5-(trichloromethyl)-1,3,4-thiadiazole,
2-[[6-deoxy-4-O-(4-O-methyl-∃-D-glycopyranosyl)-a-D-glucopyranosyl]-amino]-4-methoxy-1H-pyrrolo[2,3-d]pyrimidine-5-carbonitrile,
2-aminobutane,
2-bromo-2-(bromomethyl)-pentanedinitrile,
2-chloro-N-(2,3-dihydro-1,1,3-trimethyl-1H-inden-4-yl)-3-pyridinecarboxamide,
2-chloro-N-(2,6-dimethylphenyl)-N-(isothiocyanatomethyl)-acetamide,
2-phenylphenol (OPP),
3,4-dichloro-1-[4-(difluoromethoxy)-phenyl]-1H-pyrrol-2,5-dione,
3,5-dichloro-N-[cyano-[(1-methyl-2-propynyl)-oxy]-methyl]-benzamide,
3-(1,1-dimethylpropyl-1-oxo-1H-indene-2-carbonitrile,
3-[2-(4-chlorophenyl)-5-ethoxy-3-isoxazolidinyl]-pyridine,
4-chloro-2-cyano-N,N-dimethyl-5-(4-methylphenyl)-1H-imidazole-1-sulphonamide,
4-methyl-tetrazolo[1,5-a]quinazolin-5(4H)-one,
8-(1,1-dimethylethyl)-N-ethyl-N-propyl-1,4-dioxaspiro[4.5]decane-2-methanamine,
8-hydroxyquinoline sulphate,
9H-xanthene-2-[(phenylamino)-carbonyl]-9-carboxylic hydrazide,

bis-(1-methylethyl) 3-methyl-4-[(3-methylbenzoyl)-oxy]-2,5-thiophenedicarboxylate,
cis-1-(4-chlorophenyl)-2-(1H-1,2,4-triazol-1-yl)-cycloheptanol,
cis-4-[3-[4-(1,1-dimethylpropyl)-phenyl-2-methylpropyl]-2,6-dimethyl-morpholine hydrochloride,
ethyl [(4-chlorophenyl)-azo]-cyanoacetate,
potassium hydrogen carbonate,
methanetetrathiol sodium salt,
methyl 1-(2,3-dihydro-2,2-dimethyl-1H-inden-1-yl)-1H-imidazole-5-carboxylate,
methyl N-(2,6-dimethylphenyl)-N-(5-isoxazolylcarbonyl)-DL-alaninate,
methyl N-(chloroacetyl)-N-(2,6-dimethylphenyl)-DL-alaninate,
N-(2,3-dichloro-4-hydroxyphenyl)-1-methyl-cyclohexanecarboxamide,
N-(2,6-dimethylphenyl)-2-methoxy-N-(tetrahydro-2-oxo-3-furanyl)-acetamide,
N-(2,6-dimethylphenyl)-2-methoxy-N-(tetrahydro-2-oxo-3-thienyl)-acetamide,
N-(2-chloro-4-nitrophenyl)-4-methyl-3-nitro-benzenesulphonamide,
N-(4-cyclohexylphenyl)-1,4,5,6-tetrahydro-2-pyrimidineamine,
N-(4-hexylphenyl)-1,4,5,6-tetrahydro-2-pyrimidineamine,
N-(5-chloro-2-methylphenyl)-2-methoxy-N-(2-oxo-3-oxazolidinyl)-acetamide,
N-(6-methoxy)-3-pyridinyl)-cyclopropanecarboxamide,
N-[2,2,2-trichloro-1-[(chloroacetyl)-amino]-ethyl]-benzamide,
N-[3-chloro-4,5-bis(2-propinyloxy)-phenyl]-N'-methoxy-methanimidamide,
N-formyl-N-hydroxy-DL-alanine-sodium salt,
O,O-diethyl [2-(dipropylamino)-2-oxoethyl]-ethylphosphoramidothioate,
O-methyl S-phenyl phenylpropylphosphoramidothioate,
S-methyl 1,2,3-benzothiadiazole-7-carbothioate, and
spiro[2H]-1-benzopyran-2,1'(3'H)-isobenzofuran]-3'-one,

Bactericides:

bronopol, dichlorophen, nitrapyrin, nickel dimethyldithiocarbamate, kasugamycin, octhilinone, furancarboxylic acid, oxytetracyclin, probenazole, streptomycin, tecloftalam, copper sulphate and other copper preparations.

Insecticides / acaricide / nematicides:

abamectin, acephate, acetamiprid, acrinathrin, alanycarb, aldicarb, aldoxycarb, alpha-cypermethrin, alphamethrin, amitraz, avermectin, AZ 60541, azadirachtin, azamethiphos, azinphos A, azinphos M, azocyclotin,

Bacillus popilliae, Bacillus sphaericus, Bacillus subtilis, Bacillus thuringiensis, baculoviruses, Beauveria bassiana, Beauveria tenella, bendiocarb, benfuracarb, bensultap, benzoximate, betacyfluthrin, bifenazate, bifenthrin, bioethanomethrin, bio-permethrin, BPMC, bromophos A, bufencarb, buprofezin, butathiofos, butocarboxim, butylpyridaben,

cadusafos, carbaryl, carbofuran, carbophenothion, carbosulfan, cartap, chloethocarb, chlorethoxyfos, chlorfenapyr, chlorfenvinphos, chlorfluazuron, chlormephos, chlorpyrifos, chlorpyrifos M, chlovaporthrin, cis-resmethrin, cispermethrin, clocythrin, cloethocarb, clofentezine, cyanophos, cycloprene, cycloprothrin, cyfluthrin, cyhalothrin, cyhexatin, cypermethrin, cyromazine,

deltamethrin, demeton M, demeton S, demeton-S-methyl, diafenthiuron, diazinon, dichlorvos, diflubenzuron, dimethoat, dimethylvinphos, diofenolan, disulfoton, docusat-sodium, dofenapyn,

eflusilanate, emamectin, empenthrin, endosulfan, Entomopfthora spp., esfenvalerate, ethiofencarb, ethion, ethoprophos, etofenprox, etoxazole, etrimfos,

fenamiphos, fenazaquin, fenbutatin oxide, fenitrothion, fenothiocarb, fenoxacrim, fenoxycarb, fenpropathrin, fenpyrad, fenpyrithrin, fenpyroximate, fenvalerate, fipronil, fluazinam, fluazuron, flubrocythrinate, flucycloxuron, flucythrinate, flufenoxuron, flutenzine, fluvalinate, fonophos, fosmethilan, fosthiazate, fubfenprox, furathiocarb,

granulosis viruses,

halofenozide, HCH, heptenophos, hexaflumuron, hexythiazox, hydroprene,

imidacloprid, isazofos, isofenphos, isoxathion, ivermectin,

nuclear polyhedrosis viruses,

lambda-cyhalothrin, lufenuron

malathion, mecarbam, metaldehyde, methamidophos, Metharhizium anisopliae, Metharhizium flavoviride, methidathion, methiocarb, methomyl, methoxyfenozide, metolcarb, metoxadiazone, mevinphos, milbemectin, monocrotophos,
naled, nitenpyram, nithiazine, novaluron,
omethoat, oxamyl, oxydemethon M,
Paecilomyces fumosoroseus, parathion A, parathion M, permethrin, phenthoat, phorat, phosalone, phosmet, phosphamidon, phoxim, pirimicarb, pirimiphos A, pirimiphos M, profenofos, promecarb, propoxur, prothiofos, prothoat, pymetrozine, pyraclofos, pyresmethrin, pyrethrum, pyridaben, pyridathion, pyrimidifen, pyriproxyfen,
quinalphos,
ribavirin,
salithion, sebufos, silafluofen, spinosad, sulfotep, sulprofos,
tau-fluvalinate, tebufenozide, tebufenpyrad, tebupirimiphos, teflubenzuron, tefluthrin, temephos, temivinphos, terbufos, tetrachlorvinphos, theta-cypermethrin, thiamethoxam, thiapronil, thiatriphos, thiocyclam hydrogen oxalate, thiodicarb, thiofanox, thuringiensin, tralocythrin, tralomethrin, triarathene, triazamate, triazophos, triazuron, trichlophenidine, trichlorfon, triflumuron, trimethacarb,
vamidothion, vaniliprole, Verticillium lecanii,
YI 5302,
zeta-cypermethrin, zolaprofos,
(1R-cis)-[5-(phenylmethyl)-3-furanyl]-methyl 3-[(dihydro-2-oxo-3(2H)-furanylidene)-methyl]-2,2-dimethylcyclopropanecarboxylate,
(3-phenoxyphenyl)-methyl 2,2,3,3-tetramethylcyclopropanecarboxylate,
1-[(2-chloro-5-thiazolyl)methyl]tetrahydro-3,5-dimethyl-N-nitro-1,3,5-triazine-2(1H)-imine,
2-(2-chloro-6-fluorophenyl)-4-[4-(1,1-dimethylethyl)phenyl]-4,5-dihydro-oxazole,
2-(acetlyoxy)-3-dodecyl-1,4-naphthalenedione,
2-chloro-N-[[[4-(1-phenylethoxy)-phenyl]-amino]-carbonyl]-benzamide,
2-chloro-N-[[[4-(2,2-dichloro-1,1-difluoroethoxy)-phenyl]-amino]-carbonyl]-benzamide,

3-methylphenyl propylcarbamate.

4-[4-(4-ethoxyphenyl)-4-methylpentyl]-1-fluoro-2-phenoxy-benzene,

4-chloro-2-(1,1-dimethylethyl)-5-[[2-(2,6-dimethyl-4-phenoxyphenoxy)ethyl]thio]-3(2H)-pyridazinone,

4-chloro-2-(2-chloro-2-methylpropyl)-5-[(6-iodo-3-pyridinyl)methoxy]-3(2H)-pyridazinone,

4-chloro-5-[(6-chloro-3-pyridinyl)methoxy]-2-(3,4-dichlorophenyl)-3(2H)-pyridazinone,

Bacillus thuringiensis strain EG-2348,

[2-benzoyl-1-(1,1-dimethylethyl)-hydrazinobenzoic acid,

2,2-dimethyl-3-(2,4-dichlorophenyl)-2-oxo-1-oxaspiro[4.5]dec-3-en-4-yl butanoate,

[3-[(6-chloro-3-pyridinyl)methyl]-2-thiazolidinylidene]-cyanamide,

dihydro-2-(nitromethylene)-2H-1,3-thiazine-3(4H)-carboxaldehyde,

ethyl [2-[[1,6-dihydro-6-oxo-1-(phenylmethyl)-4-pyridazinyl]oxy]ethyl]-carbamate,

N-(3,4,4-trifluoro-1-oxo-3-butenyl)-glycine,

N-(4-chlorophenyl)-3-[4-(difluoromethoxy)phenyl]-4,5-dihydro-4-phenyl-1H-pyrazole-1-carboxamide,

N-[(2-chloro-5-thiazolyl)methyl]-N'-methyl-N''-nitro-guanidine,

N-methyl-N'-(1-methyl-2-propenyl)-1,2-hydrazinedicarbothioamide,

N-methyl-N'-2-propenyl-1,2-hydrazinedicarbothioamide,

O,O-diethyl [2-(dipropylamino)-2-oxoethyl]-ethylphosphoroamidothioate.

A mixture with other known active compounds, such as herbicides, or with fertilizers and growth regulators is also possible.

Furthermore, when used as nematicides, the active compounds according to the invention can be present in their commercial formulations and in the use forms, prepared from these formulations, as a mixture with synergists. Synergists are compounds which increase the action of the active compounds, without it being necessary for the synergist added to be active itself.

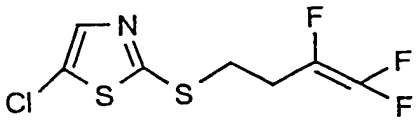
The active-compound content of the use forms prepared from the commercial formulations can vary within wide limits. The active-compound concentration of the use forms can be from 0.0000001 to 95% by weight of active compound, preferably between 0.0001 and 1% by weight.

Application is carried out in a customary manner adapted to the use forms.

The preparation and the use of the compounds according to the present invention will be described more specifically by the following examples. However, the present invention should not be restricted to them in any way. "Parts" mean "parts by weight" unless specified otherwise.

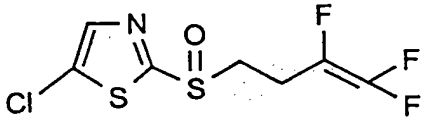
Preparation Examples

Example 1



2-(3,4,4-Trifluoro-3-butenylthio)thiazole (6.75 g, 30 mM) is dissolved in carbon tetrachloride (60 ml). N-chlorosuccinimide (4.8 g) is added to the solution and refluxed for 18 hours by heating. As soon as the reaction has reached room temperature, the mixture is filtered and the solvent is distilled off. The concentrate is purified by column chromatography (eluent: hexane/ethyl acetate = 90/10) to obtain 5-chloro-2-(3,4,4-trifluoro-3-butenylthio)thiazole as pale yellow liquid (n^{20}_D 1.5326).

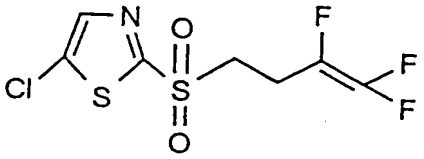
Example 2



5-Chloro-2-(3,4,4-trifluoro-3-butenylthio)thiazole (2.07 g, 8 mM) is dissolved in chloroform (40 ml). m-chloroperoxybenzoic acid (1.38 g) is added to the solution under ice cooling (temperature below 4°C) and further stirred for 8 hours at a temperature below 4°C.

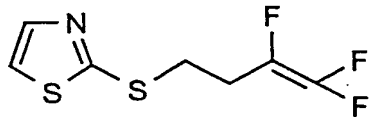
10% sodium thiosulfate is added to the solution and the solution is then fractionated. The chloroform layer is washed with 5% aqueous solution of sodium hydroxide and dried over unhydrous magnesium sulfate. The solvent is distilled off and the concentrate is purified by column chromatography (eluent: hexane/ethyl acetate = 90/10) to obtain 5-chloro-2-(3,4,4-trifluoro-3-butenylsulfinyl)thiazole (1.5 g) as pale yellow liquid (n^{20}_D 1.5380).

Example 3



To the solution of 5-chloro-2-(3,4,4-trifluoro-3-butenylthio)thiazole (2.60 g, 10 mM) and acetic acid (28 g) 31% hydrogen peroxide water (3.29 g) is added and stirred at 55-60°C for 6 hours. After cooling to 5°C the reaction mixture is adjusted to pH 6 by adding an appropriate amount of an aqueous solution of sodium hydroxide, diluted with water and extracted three times with chloroform (25 ml). The chloroform layer is washed with water, 10% sodium thiosulfate and water in this order, and dried over unhydrous sodium sulfate. The solvent is distilled off and the concentrate is purified by column chromatography (eluent: hexane/ethyl acetate = 90/10) to obtain 5-chloro-2-(3,4,4-trifluoro-3-butenylsulfonyl)thiazole (2.2 g) as pale yellow liquid (n^{20}_D 1.5205).

Reference Example



2-Mercaptothiazole (5.18 g), potassium carbonate (6.72 g) and 4-bromo-1,1,2-trifluorobutene-1 (9.21 g) are refluxed in acetonitrile (60 ml) in the presence of argon gas for 6 hours by heating. After the reaction mixture has reached room temperature, it is filtered and the solvent is distilled off. The residue is dissolved in dichloromethane and washed with 5% aqueous solution of sodium hydroxide and water in this order. It is dried over unhydrous sodium sulfate and purified by column chromatography (eluent: dichloromethane) to obtain 2-(3,4,4-trifluoro-3-butenyl-thio)thiazole (8.6 g) as pale yellow liquid (n^{20}_D 1.5200).

Use Examples

Example 1 Test against Meloidogyne spp. (Soil pot test)

Preparation of test agent:

1 Part of the active compound is impregnated to 99 parts of pumice to obtain fine granules.

Test method:

The test agent prepared as mentioned above was added to soil contaminated with Meloidogyne incognita to a chemical concentration of 10 ppm and homogeneously mixed by stirring. A pot (1/5000 are) was filled with the soil. About 20 seeds of tomato (variety: Kurihara) were sown per pot. After cultivation in a greenhouse for 4 weeks, they were carefully pulled out not to damage the roots and the root knot index and the controlling effect were determined as follows.

Degree of damage
0: No knots were formed (Complete control).
1: A few knots were formed.
2: Knots were formed to a medium extent.
3: Knots were formed to an intense extent.
4: Knots were formed to the most intense extent (which corresponds to non-treatment).

$$\text{Root knot index} = \frac{\Sigma\ (\text{degree of damage} \times \text{number of individuals})}{\text{Total number of tested individuals} \times 4} \times 100$$

The controlling effect of the compounds tested can then be evaluated according to the following equation:

$$\text{Controlling effect [\%]} = \frac{(\text{Root knot index at non-treated area} - \text{Root knot index at treated area})}{\text{Root knot index at non-treated area}} \times 100$$

The evaluation of the controlling effects of the compounds according to the present invention was done on the basis of the values of the controlling effect which can be obtained in the above-mentioned way and were connected with the following standards:

a: Controlling effect 100-71%
b: Controlling effect 70-50%
c: Controlling effect less than 50%
d: Controlling effect 0%

Results are shown in the following Table 1.

Table 1

Compound Ex. No.	Concentration of active ingredient [ppm]	Evaluation of controlling effect
1	10	a
2	10	a
3	10	a

Formulation Examples

Example 1 (Granule)

To a mixture of 10 parts of a compound according to the present invention (Example No. 1), 30 parts of bentonite (montmorillonite), 58 parts of talc and 2 parts of ligninsulphonate salt, 25 parts water are added, well kneaded, worked up into granules of 10-40 mesh with the help of an extrusion granulator and dried at 40-50°C to obtain granules.

Example 2 (Granule)

95 Parts of clay mineral particles having a particle diameter distribution of 0.2-2 mm are put into a rotary mixer. While rotating it, 5 parts of a compound according to the present invention (Example No. 2) are sprayed onto the mineral particles together with a liquid diluent to obtain uniformly wetted particles and the particles are then dried at 40-50°C to obtain granules.

Example 3 (Emulsifiable concentrates)

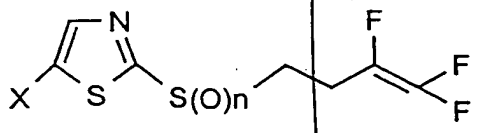
30 Parts of a compound according to the present invention (Example No. 3), 55 parts of xylene, 8 parts of polyoxyethylene alkyl phenyl ether and 7 parts of calcium alkylbenzenesulphonate are mixed and stirred to obtain an emulsion.

Example 4 (Wettable powder)

15 parts of a compound according to the present invention (Example No. 1), 80 parts of a mixture of white carbon (hydrous amorphous silicon oxide fine powders) and powder clay (1:5), 2 parts of sodium alkylbenzenesulphonate and 3 parts of sodium alkylnaphthalenesulphonate-formalin-condensate are crushed and mixed together to obtain a wettable powder.

Claims

1. Compounds of the formula (I)



(I),

wherein

X represents halogen, and

n represents 0, 1 or 2.

2. Compounds of the formula (I) according to claim 1, wherein

X represents fluoro, chloro or bromo, and

n represents 0 or 2.

3. Compounds of the formula (I) according to claim 1 or claim 2, wherein

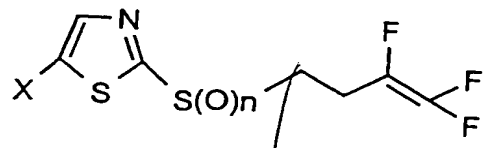
X represents chloro or bromo, and

n represents 2.

4. Compounds of the formula (I) according to claims 1 to 3, wherein

X represents chloro.

5. Process for preparing compounds of the formula (I)



(I)

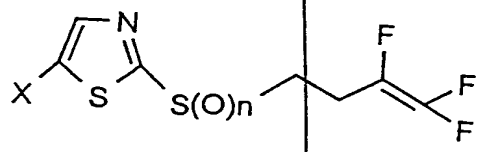
wherein

X is as defined in claims 1 to 4, and

n represents 0,

characterized in that 2-(3,4,4-trifluoro-3-butenylthio)thiazole is reacted with a halogenating agent, if appropriate in the presence of inert solvents.

6. Process for preparing compounds of the formula (I)

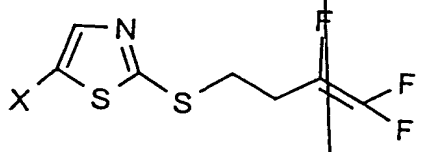


(I)

wherein

n represents 1 or 2, and

X is as defined in claims 1 to 4, characterized in that compounds of the formula (Ib)



(Ib)

wherein

X is as defined in claims 1 to 4.

are reacted with an oxidizing agent, if appropriate in the presence of inert

solvents.

7. Nematicidal compositions, characterized in that they contain at least one compound of the formula (I) according to claims 1 to 4.

8. A method of combating nematodes, characterized in that compounds of the formula (I) according to claims 1 to 4 are allowed to act on nematodes and/or their habitat.

9. Use of the compounds of the formula (I) according to claims 1 to 4 for combating nematodes.

10. Process for preparing nematicidal compositions, characterized in that the compounds of the formula (I) according to claims 1 to 4 are mixed with extenders and/or surface active agents.

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference	FOR FURTHER ACTION	See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)
NIT 364-WO Lu		
International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/IB00/00868	28/06/2000	06/07/1999
International Patent Classification (IPC) or national classification and IPC C07D277/36		
Applicant NIHON BAYER AGROCHEM K.K. et al.		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 5 sheets, including this cover sheet.

   ☐ This report is also accompanied by ANNEXES, i.e. sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

   These annexes consist of a total of sheets.

3. This report contains indications relating to the following items:

   - I ☒ Basis of the report
   - II ☐ Priority
   - III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
   - IV ☐ Lack of unity of invention
   - V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations suporting such statement
   - VI ☐ Certain documents cited
   - VII ☒ Certain defects in the international application
   - VIII ☐ Certain observations on the international application

Date of submission of the demand	Date of completion of this report
01/12/2000	24.09.2001
Name and mailing address of the international preliminary examining authority: European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized officer Usuelli, A Telephone No. +49 89 2399 7366

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/IB00/00868

I. Basis of the report

1. With regard to the **elements** of the international application *(Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rules 70.16 and 70.17))*:
Description, pages:

1-21 as originally filed

Claims, No.:

1-10 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:
- ☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

(Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-10
	No:	Claims	
Inventive step (IS)	Yes:	Claims	
	No:	Claims	1-10
Industrial applicability (IA)	Yes:	Claims	1-10
	No:	Claims	

2. Citations and explanations
see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

Re Item V

Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1- In the present Written Opinion reference is made to the following documents:

d1: WO-A-8607590
d2: WO-A-9524403

2- Novelty
D1 discloses a broad class of trifluorobutenyl derivatives which encompasses also the compounds of the invention when R (in d1) is an "optionally substituted thiazolinyl". However there are neither specific examples nor subclasses of preferred compounds in which said group R represent a 5-halogen-thiazolinyl. The present compounds of formula (I) are therefore taken as a novel selection over the compounds of d1.
The compounds of the invention are novel vis-à-vis d2 on account of the fluorine atom on the second carbon atom of the butenyl moiety.
Accordingly, claims 1-10 fulfil the requirements of Art. 33(2) PCT.

3- Inventive step
3.1- Both d1 and d2 disclose compounds useful for combatting infestations of nematodes. The compound of example 16 of d1 (page 22) differs from the compounds of the invention (cf. in particular compound of example 1) only in that the hydrogen in position 5 of the thiazole ring has been replaced by a halogen. The closest compounds of d2 are the compounds VII.24-VII.26 (page 16) which differ from the compounds of the invention (cf. in particular the compounds of examples 1-3) only in that they have a hydrogen atom on the C-2 of the butenyl moiety instead of a fluorine. This means that, starting from the closest prior art, the technical problem of providing further nematicides has been solved by introducing in the known compounds a halogen in replacement of a hydrogen. It appears that the skilled man would not be surprised that this minimal modification of the prior art compounds would allow to obtain further compounds useful as nematicides. This is indirectly demonstrated by the fact that those compounds of d1 and d2 which differ from each other only in that a halogen replaces a hydrogen (cf. for instance the compound of example 16 of d1 with the compound VII.1 of d2), have the same activity.

3.2- Accordingly, an inventive step could be acknowledged only if it is shown by means of comparative tests that the compounds of the invention have unexpected effects vis-à-vis the compounds of d1 and d2. In order to have significant results, the compounds compared should have the maximum degree of similarity (for instance the compound 16 of d1 must be compared with the compound of the example 1 of the present application).

3.3- In view of the above paragraphs claims 1-10 do not fulfil the requirements of Art. 33(3) PCT.

Re Item VII

Certain defects in the international application

1- The units of pressure "bar" employed on pages 4 line 29 and 5 line 29, are not additionally expressed in terms of the units stipulated by Rule 10.1/(a)/and/(b) PCT.

2- The sentences "However...in any way" (page 16, lines 8-9) is considered irrelevant and thus superfluous, cf. Rule 9.1 (iv) PCT.

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference	**FOR FURTHER ACTION** see Notification of Transmittal of International Search Report (Form PCT/ISA/220) as well as, where applicable, item 5 below.	
NIT 364-WO		
International application No.	International filing date *(day/month/year)*	(Earliest) Priority Date *(day/month/year)*
PCT/IB 00/ 00868	28/06/2000	06/07/1999
Applicant		
NIHON BAYER AGROCHEM K.K.		

This International Search Report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This International Search Report consists of a total of 2 sheets.

[X] It is also accompanied by a copy of each prior art document cited in this report.

1. **Basis of the report**

   a. With regard to the **language,** the international search was carried out on the basis of the international application in the language in which it was filed, unless otherwise indicated under this item.

   [ ] the international search was carried out on the basis of a translation of the international application furnished to this Authority (Rule 23.1(b)).

   b. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international search was carried out on the basis of the sequence listing :

   [ ] contained in the international application in written form.

   [ ] filed together with the international application in computer readable form.

   [ ] furnished subsequently to this Authority in written form.

   [ ] furnished subsequently to this Authority in computer readble form.

   [ ] the statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.

   [ ] the statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished

2. [ ] **Certain claims were found unsearchable** (See Box I).

3. [ ] **Unity of invention is lacking** (see Box II).

4. With regard to the **title,**

   [X] the text is approved as submitted by the applicant.

   [ ] the text has been established by this Authority to read as follows:

5. With regard to the **abstract,**

   [X] the text is approved as submitted by the applicant.

   [ ] the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box III. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority.

6. The figure of the **drawings** to be published with the abstract is Figure No. -

   [ ] as suggested by the applicant.

   [ ] because the applicant failed to suggest a figure.

   [ ] because this figure better characterizes the invention.

   [ ] None of the figures.

INTERNATIONAL SEARCH REPORT

International Application No
PCT/IB 00/00868

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 C07D277/36 A01N43/78

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 C07D A01N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, CHEM ABS Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 86 07590 A (FMC CORPORATION) 31 December 1986 (1986-12-31) cited in the application the whole document, particularly example 16	1-10
Y	WO 95 24403 A (ZENECA LIMITED) 14 September 1995 (1995-09-14) cited in the application the whole document, particularly pages 15-19, table VII, and examples VII.1-VII.16	1-10

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 September 2000	29/09/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Allard, M

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/IB 00/00868

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 8607590 A	31-12-1986	AU 601656 B	13-09-1990
		AU 6122986 A	13-01-1987
		BR 8606746 A	13-10-1987
		CA 1277668 A	11-12-1990
		CN 86104207 A	01-04-1987
		DK 84387 A	19-02-1987
		EP 0228447 A	15-07-1987
		HU 42424 A	28-07-1987
		HU 204022 B	28-11-1991
		OA 8484 A	29-07-1988
		ZA 8604637 A	24-02-1988
		JP 63500037 T	07-01-1988
		MX 9249 A	01-12-1993
		US 4952580 A	28-08-1990
WO 9524403 A	14-09-1995	AP 649 A	29-05-1998
		AU 685242 B	15-01-1998
		AU 1816495 A	25-09-1995
		BG 100900 A	31-07-1997
		BR 9507042 A	09-09-1997
		CA 2182520 A	14-09-1995
		CN 1143958 A	26-02-1997
		CZ 9602632 A	11-12-1996
		EG 20790 A	29-03-2000
		EP 0749433 A	27-12-1996
		FI 963539 A	09-09-1996
		HU 74902 A,B	28-02-1997
		JP 9510197 T	14-10-1997
		LV 11686 A,B	20-02-1997
		NO 963776 A	07-11-1996
		NZ 281267 A	24-11-1997
		PL 316175 A	23-12-1996
		SK 114896 A	09-04-1997
		US 5705516 A	06-01-1998
		US 5912243 A	15-06-1999